CMP
United Business Media

PRODUCT REVIEWS: ADOBE PHOTOSHOP 7, 3DLABS' WILDCAT VP

JANUARY 2003

GameDeveloper

ON THE FRONT LINE OF GAME INNOVATION

Presenting...

The 2002 Front Line Awards

10 Of The Year's Best Development Tools

Hiring Screenwriters For Your Game

A Good Idea?

Postmortem:

Monolith's

NO ONE LIVES FOREVER 2

$5.95US $6.95CAN
01
0 71486 02133 9
Display Until January 24, 2003

3ds max

Every day, gamers buy more games created with **3ds max** than any other animation software. Which makes sense when you consider **3ds max** has helped create thousands of games in every genre and platform. Whether it's RPG, Strategy, Simulation or good clean fun plundering Glukkons, all **3ds max** games have one thing in common: Gamers can't seem to get enough.

discreet™

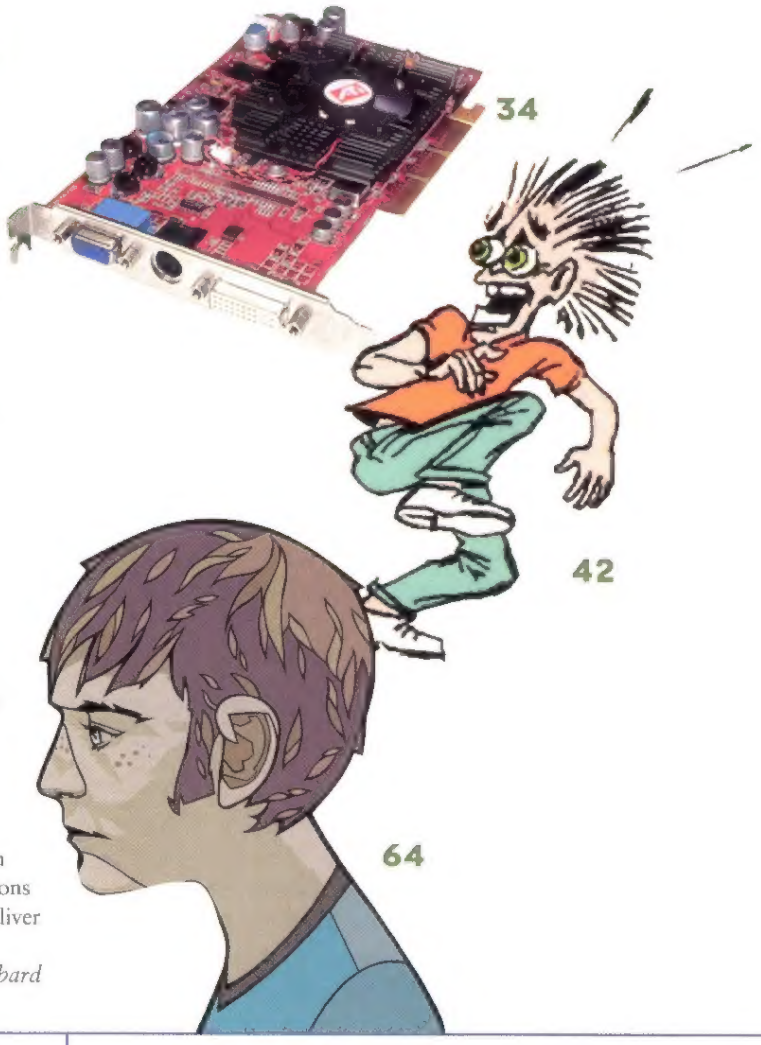
For more info on **3ds max** and other 3D products like **gmax**® and **character studio**®, call 1.800.869.3504 or go to www.discreet.com.

Discreet is a division of Autodesk, Inc. Discreet, character studio, 3ds max, and reactor are either registered trademarks or trademarks of Autodesk, Inc. in the U.S.A. and/or other countries. All other brand names, product names or trademarks belong to their respective holders. ©2002 Autodesk, Inc. All rights reserved. ©1987, 2001 Konami Computer Entertainment Japan, Planet Moon Studios, Oddworld: Munch's Oddysee ©2002 Oddworld Inhabitants, Inc. All Rights Reserved. The Oddworld logo, Oddworld: Munch's Oddysee are either registered trademarks or trademarks of Oddworld Inhabitants in the United States and/or other countries. Lionhead Studios/Electronic Arts™, Diablo® II box cover image courtesy of Blizzard Entertainment®, Remedy Entertainment & 3D Realms Entertainment, ©1999-2002 Activision, Inc. and its affiliates. Activision is a registered trademark and Activision 02, Tony Hawk's Pro Skater and Pro Skater are trademarks of Activision, Inc. and its affiliates. All Rights Reserved. Tony Hawk is a trademark of Tony Hawk. The ratings icon is a trademark of the Interactive Digital Software Association. Image courtesy of Electronic Arts™, 2002, ©Microsoft Corporation. All rights reserved. Microsoft, Bungie, Halo, Xbox and the Xbox logo are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries. Valve Software/Sierra.

CONTENTS

VOLUME 10, NUMBER 1

January 2003



FEATURES

34 The 2002 Front Line Awards

For the fifth straight year, *Game Developer* brings you the best of the best as selected by our panel of industry professionals. Which game development software and hardware merit the distinction of our Front Line Awards? Find out inside.

Daniel Huebner

42 What Screenwriters Don't Know About Games

A long list of television shows and movie scripts does not necessarily qualify a Hollywood screenwriter to write the dialogue for your game. Freeman explains how to help the writer adapt to your project's needs.

David Freeman

48 Postmortem: Monolith's No One Lives Forever 2

Making a good sequel to a hit game is a challenge on many levels. Find out how Monolith applied the lessons learned on the original NO ONE LIVES FOREVER to deliver a critically acclaimed sequel on-time and on-budget.

Craig Hubbard

DEPARTMENTS

4 **GAME PLAN** *Jennifer Olsen*
Tales from the Front Line

6 **SAYS YOU**
Readers write back

8 **INDUSTRY WATCH** *Everard Strong*
Nintendo pays, Nokia plays, G4 expands, more

11 **PRODUCT REVIEWS**
Adobe's Photoshop 7.0, 3Dlabs' WildcatVP

14 **PROFILES** *Everard Strong*
American McGee talks about Hollywood's production methodology, and why games are beyond that.

COLUMNS

16 **THE INNER PRODUCT** *Jonathan Blow*
Interactive Profiling, Part 2: Behavior Analysis

22 **ARTIST'S VIEW** *Hayden Duvall*
What's Wrong with Our Games?

28 **SOUND PRINCIPLES** *Mike Verrette*
Voices of Reason

30 **BETTER BY DESIGN** *Noah Falstein*
AI Without Pain

64 **SOAPBOX** *Gerard Jones*
Playing It Out

COVER: *The cover was created exclusively for* Game Developer *by Monolith senior artist Scott Albaugh and art director David Longo. Cate Archer was modeled, posed, and lit in Maya. The full render was then brought into Photoshop, where the final image was touched up and composited with the background design.*

Game Developer (ISSN 1073-922X) is published monthly by CMP Media LLC, 600 Harrison St., San Francisco, CA 94107, (415) 947-6000. Please direct advertising and editorial inquiries to this address. Canadian GST: 124513185. **SUBSCRIPTION RATES:** Subscription rate for the U.S. is $49.95 for twelve issues. Countries outside the U.S. must be prepaid in U.S. funds drawn on a U.S. bank or via credit card. Canada/Mexico: $69.95; all other countries: $99.95 (issues shipped via air delivery). Periodical postage paid at San Francisco, CA and additional mailing offices. **POSTMASTER:** Send address changes to Game Developer, P.O. Box 1274, Skokie, IL 60076-8274. **CUSTOMER SERVICE:** For subscription orders and changes of address, call toll-free in the U.S. (800) 250-2429 or fax (847) 647-5972. All other countries call (1) (847) 647-5928 or fax (1) (847) 647-5972. Send payments to *Game Developer*, P.O. Box 1274, Skokie, IL 60076-8274. For back issues write to *Game Developer*, 1601 W. 23rd St. Ste. 200, Lawrence, KS 66046-2703. Call toll-free in the U.S./Canada (800) 444-4881 or fax (785) 841-2624. All other countries call (1) (785) 841-1631 or fax (1) (785) 841-2624. Please remember to indicate *Game Developer* on any correspondence.

RenderWare

Studio

RenderWare Studio is a unique collaborative game production environment that allows the whole team to create, view and tune games in real-time on multiple platforms.

RenderWare Studio encompasses the entire development process from pre-production to QA, allowing all team members to work in parallel and remove the time-consuming bottlenecks that can occur.

Artist Environment

Art tools & modelers such as 3ds max, Maya, Softimage

ART EXPORT

CODE EXPORT

Programmer Environment

Compilers and debug tools, such as CodeWarrior, Visual Studio, SN ProDG, plus custom tools from RenderWare Studio SDK

Platform

RenderWare Platform is the most comprehensive suite of development tools and technologies available, seamlessly integrating Graphics, Audio, AI and Physics.

With v3.4, the industry standard in multi-platform middleware is now more powerful than ever before, offering unrivalled performance – *more than twice the power of other graphics solutions.*

Graphics

- High performance, low footprint, easily customized libraries with PowerPipe™ API
- Comprehensive art-path tool chain
- Powerful exporter framework
- Cutting edge, genre specific FX Packs
- Target-specific enhancements
- dPVS plugin available (powered by Hybrid)

We are proud to be sponsors of...

GameDeveloper FRONTLINE FINALIST '02

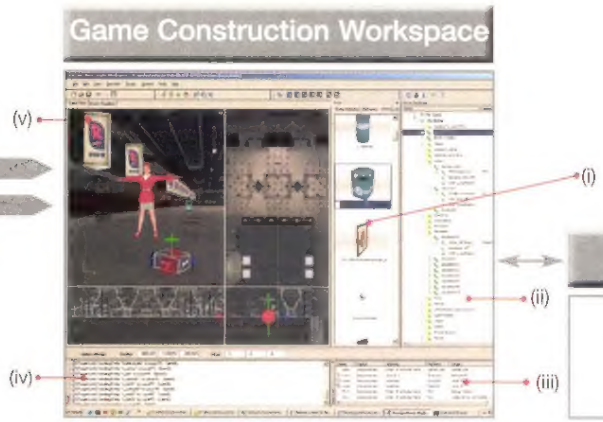
igda international game developers association

game developers CHOICE awards

For further information please visit www.renderware.com, email rw-info@csl.com, or call 512-478-5605

We're proud to announce RenderWare is now being used in hundreds of games and would like to congratulate our customers on generating over $ 1 BILLION of revenue from the titles that have been released by the end of 2002! Here's to a great 2003!

>>Powerful, Intuitive, Fast...



(i) Object Lister
(ii) World Lister
(iii) Console Link
(iv) Output Window
(v) Game View

Asset Management

Powered by alienbrain®

1 The level designer downloads and executes the actual game to the target console using RenderWare Studio. A dynamic link is maintained for full interaction.

2 Source Code is parsed by RenderWare Studio to generate the UI that controls chosen game parameters exposed to the designer.

3 Objects can be drag 'n' dropped from the asset window into the game window. Behaviors can then be dropped onto game geometry.

4 Object/behavior combinations are then instantly available for tuning gameplay in the attribute editor window.

Network Link

Game Application

Audio

- Dolby Digital & Pro Logic 1 & 2
- Outstanding multi-platform real-time audio
- Comprehensive sound bank management
- Virtual voice management

A.I.

- Powerful character & racing behaviors
- Multi-genre core capabilities
- Custom agent & brain scripting
- Powerful 3D pathfinding
- Time-slicing

Powered by kynogon

Physics

- Full rigid body dynamics
- Multiple joint types including constrained ball & socket
- Fast friction modeling
- Ragdoll and vehicle optimizations
- Small memory footprint
- Time-slicing

Engineered by KARMA

RenderWare Core

XBOX · PlayStation 2 · NINTENDO GAMECUBE · PC

>> Get To The *Game*

Criterion SOFTWARE

© 2002 Criterion Software Inc. All rights reserved. Criterion and RenderWare are registered trademarks of Canon Inc. RenderWare Studio screenshot courtesy of Rage Software plc © 2002. PlayStation is a registered trademark of Sony Computer Entertainment Inc. Microsoft is a registered trademark and Xbox is a trademark of Microsoft Corporation. Nintendo is a registered trademark and NINTENDO GAMECUBE is a trademark of Nintendo Co., Ltd. All other trademarks mentioned herein are the property of their respective companies.

GAME PLAN

LETTER FROM THE EDITOR

Tales from the Front Line

This issue marks the fifth installment of *Game Developer*'s annual Front Line Awards. Every year the job of picking the year's best game development tools gets easier, because there are more quality tools for developers to choose from, and every year picking them gets harder, because there are more quality tools for developers to choose from.

Our panel of industry professionals who judge the year's releases face the same challenges you do in their daily work, and they want the same things from their development tools: solutions, stability, and support. Developers who used to have to sacrifice stability in their tools for feature richness, for example, now have a better chance of getting both.

Vendors who have listened to game developers' requests and feedback over the years are being rewarded for their efforts to meet this industry's unique needs in functionality and support, at a time when business in related fields such as broadcast, film, and web production has declined for many of them.

Even vendors who lacked a revision release within the 2002 Front Line Awards' eligibility window of September 1, 2001, to August 31, 2002 (including one of the 900-pound gorillas of the Art category, Maya), had a significant impact on the industry this year with price cuts. The ramifications in the 3D software business alone were huge.

At the beginning of the high-tech downturn, vendors approached the game development industry to find that game developers were not only buying, their budgets were actually rising rapidly during the last console transition. That rise is now leveling off. Most developers currently undertaking new projects are doing so with similar and in some cases even slightly smaller budgets. And while a slowdown in budget growth might seem a dismal picture for vendors serving the game development market, there remains plenty of opportunity for vendors who understand that changes in how developers are allocating their budget resources are ultimately far more significant to their business than any hiccups in how much they are spending.

The role of NIH syndrome is becoming largely ceremonial at many game development studios today, thanks to the growing technical and budgetary arguments in favor of the increasing range of workable middleware options available, which are represented in this year's Front Line Awards in the new Game Components category. The Production category is on hiatus this year — frankly, we were not sufficiently impressed with the field of eligible nominees to give any an award in this underserved area, unless we felt like honoring Microsoft Excel, which hardly fits the awards' goal of recognizing products that specifically address game development needs. Last year's winner in the category, NXN Alienbrain, has since added so much software-configuration management functionality (including a bundled version of Araxis Merge, a 2001 Front Line Award winner), it may find a home in the Programming category next year when version 6, released after this year's eligibility cut-off, will be eligible.

It's a poor workman who blames his tools, says the old adage, and it's as true in game development as it is in carpentry or anything else. Nevertheless, developers have been stymied in the past by some truly inadequate development tools. Today, more vendors than ever understand and appreciate the strategic importance of enabling game developers to meet the challenges of delivering top-quality work within a reasonable budget. With the Front Line Awards, we want to recognize those vendors and products, and encourage them to continue listening to their game development customers, so they might serve this market even better in the future to the benefit of all.

Jennifer Olsen
Editor-In-Chief

GameDeveloper

600 Harrison Street, San Francisco, CA 94107 t: 415.947.6000 f: 415.947.6090

Publisher
Jennifer Pahlka *jpahlka@cmp.com*

EDITORIAL

Editor-In-Chief
Jennifer Olsen *jolsen@cmp.com*

Managing Editor
Everard Strong *estrong@cmp.com*

Production Editor
Olga Zundel *ozundel@cmp.com*

Product Review Editor
Daniel Huebner *dan@gamasutra.com*

Art Director
Audrey Welch *awelch@cmp.com*

Editor-At-Large
Chris Hecker *checker@d6.com*

Contributing Editors
Jonathan Blow *jon@number-none.com*
Hayden Duvall *hayden@confounding-factor.com*
Noah Falstein *noah@theinspiracy.com*

Advisory Board
Hal Barwood LucasArts
Ellen Guon Beeman Monolith
Andy Gavin Naughty Dog
Joby Otero Luxoflux
Dave Pottinger Ensemble Studios
George Sanger Big Fat Inc.
Harvey Smith Ion Storm
Paul Steed Independent

ADVERTISING SALES

Director of Sales/Associate Publisher
Michele Sweeney e: *msweeney@cmp.com* t: 415.947.6217

Senior Account Manager, Eastern Region & Europe
Afton Thatcher e: *athatcher@cmp.com* t: 415.947.6224

Account Manager, Northern California & Southeast
Susan Kirby e: *skirby@cmp.com* t: 415.947.6226

Account Manager, Recruitment
Raelene Maiben e: *rmaiben@cmp.com* t: 415.947.6225

Account Manager, Western Region & Asia
Craig Perreault e: *cperreault@cmp.com* t: 415.947.6223

Account Representative
Aaron Murawski e: *amurawski@cmp.com* t: 415.947.6227

ADVERTISING PRODUCTION

Vice President, Manufacturing Bill Amstutz
Advertising Production Coordinator Kevin Chanel
Reprints Cindy Zauss t: 909.698.1780

GAMA NETWORK MARKETING

Director of Marketing Greg Kerwin
Senior MarCom Manager Jennifer McLean
Marketing Coordinator Scott Lyon

CIRCULATION

BPA
Game Developer is BPA approved

Group Circulation Director Catherine Flynn
Circulation Manager Ron Escobar
Circulation Assistant Ian Hay
Newsstand Analyst Pam Santoro

SUBSCRIPTION SERVICES

For information, order questions, and address changes
t: 800.250.2429 or 847.647.5928 f: 847.647.5972
e: *gamedeveloper@halldata.com*

INTERNATIONAL LICENSING INFORMATION

Mario Salinas
t: 650.513.4234 f: 650.513.4482 e: *msalinas@cmp.com*

CMP MEDIA MANAGEMENT

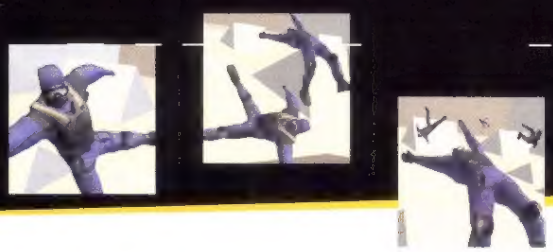
President & CEO Gary Marshall
Executive Vice President & CFO John Day
Chief Operating Officer Steve Weitzner
Chief Information Officer Mike Mikos
President, Technology Solutions Group Robert Faletra
President, Healthcare Group Vicki Masseria
President, Electronics Group Jeff Patterson
President, Specialized Technologies Group Regina Starr Ridley
Senior Vice President, Global Sales & Marketing Bill Howard
Senior Vice President, HR & Communications Leah Landro
Vice President & General Counsel Sandra Grayson
Vice President, Creative Technologies Philip Chapnick

CMP
United Business Media

GamaNetwork

WWW.GAMANETWORK.COM

Hiring now



Havok 2.
Physics for characters.
Available now.

Create unprecedented physical effects with multiple real-time ragdolls. Check out our demos at www.havok.com/havok2

havok

We supply the physics, you supply the imagination.

GameDeveloper FRONTLINE FINALIST '02

SAYS YOU

A FORUM FOR YOUR POINT OF VIEW. GIVE US YOUR FEEDBACK...

Storytelling Pitfalls

I was relating "The Godfather Paradox" (Better by Design, November 2002) to a friend in the industry when it occurred to me that movies and other media that tell stories are one of the few kinds of products that can get worse on successive iterations. If you compare videogames to, say, vacuum cleaners or new models of cars, you'd expect them to get better each time around. It seems like Noah Falstein makes the classic mistake of comparing videogames to movies. I think it's better to think of a videogame "sequel" as an improvement on the same product.

As ZELDAS get re-released, they have the tendency to tell an approximation of the same story over and over again. ZELDA: OCARINA OF TIME is not a sequel to ZELDA: LINK TO THE PAST in the narrative sense; it feels more like another draft of the same story. It's better to compare it to Hollywood remakes than sequels. (I, personally, liked John Carpenter's version of *The Thing* much better than the original, although I know there are critics out there who bash remakes.)

I'll go out on a limb here: Maybe trying to pick up a narrative thread where METAL GEAR SOLID left off is one of the places where MGS2 got in trouble? If Hideo Kojima had instead rehashed the METAL GEAR story — improving upon its flaws — maybe MGS2 sales would be overtaking MGS now.

We need more people like Falstein shining light into the most important and least understood area of videogames. I fear game design, like art, is resistant to rational analysis. Still, rational analysis is one of the only tools at our disposal.

Jamie Fristrom
Treyarch
via e-mail

NOAH FALSTEIN RESPONDS: *I think that it's a little deceptive to compare games to vacuum cleaners. I'd be more interested in seeing other forms of entertainment that people expect to get better in sequels. And I'm not sure you can even properly consider a game like* ZELDA: OCARINA OF TIME *a remake; games with familiar characters, unlike old 3D arcade games, just feel different. But it all comes down to apples and oranges, and you're absolutely right that it's dangerous to push the comparisons of different media too far.*

I'm intrigued by your observation that storytelling does poorly with sequels. If you consider characters being reused, things change quite a bit — you get an explanation for the popularity of sitcoms or even TV dramas, where characters seem to matter more than telling really new stories. For that matter, the James Bond or Dirty Harry movies come to mind, or other action-hero films where the actor plays a similar role in each film even when the stories and even characters are supposedly different.

What About Brazilians?

I'm the CIO of a game development company in Brazil. I enjoyed Jennifer Olsen's editorial "Hello World" (Game Plan, November 2002). The only omission I saw was that you said nothing about the game market in Brazil or South America. Although games from that region don't have much of a presence in the global market today, the industry is growing rapidly there.

In fact, DEER HUNTER 3 was made by a Brazilian company, Southlogic Studios, and released by Infogrames.

Giuliano B. Schiavon
Oniria Entertainment
via e-mail

Entrepreneurs Unite!

I enjoyed "Games for the People, by the People" (Soapbox, October 2002), in which Matthew Stibbe wrote ". . . it would be exciting to see the open source movement do more to embrace games."

I've long wanted to make it possible for ordinary people (like myself) to create games; and so, as part of the Exult project, I've written a map editor and script compiler. My goal is to allow nonprogrammers to create large, plot-oriented RPGs.

The problem, which I imagine most professional game developers already know, is that creating a game is far more than programming or plot development.

Jeff Freedman
via e-mail

Ethics, Shmethics

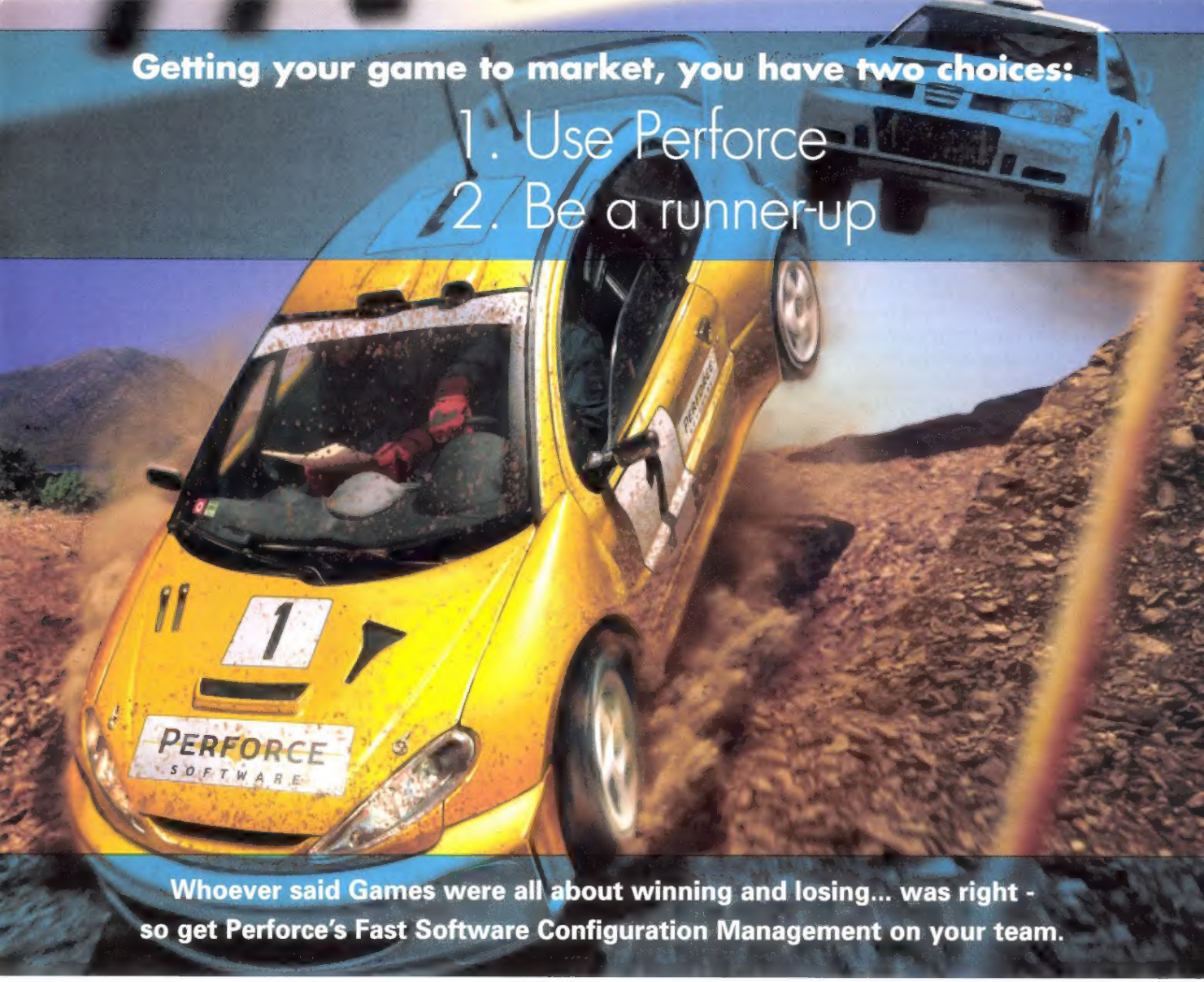
I recently read Jennifer Olsen's editorial "Decisions, Decisions" (Game Plan, September 2002). With the media attention GRAND THEFT AUTO 3 has been receiving, the analysis has been entirely focused on the metaphor of the game, which is relevant to the marketing of the product, but not overly relevant to any analysis of lasting effects of the game on a player.

The message of GTA (the game) itself is much less inflammatory than seems implied by either a description of the game metaphor or the media attention it has received. After playing heavily for a couple of weeks, the only residual game effect I experienced in real life was an increased awareness of police cars while walking down the street; not a big deal, well within the constraints of reasonable human behavior, and in some places in the world, enhancement of a positive survival trait.

Emergent gameplay is about giving the user the freedom to explore his own ethical quandaries, not imposing our own upon them. Any game that tries to model such ethical content will succeed or fail not on the specific metaphorical nature of the quandary, but on the resulting gameplay. The albatross is a development quality thing, not an ethics of metaphor thing.

Gregor Koomey
via e-mail

Let us know what you think: send us an e-mail at editors@gdmag.com, or write to *Game Developer*, 600 Harrison St., San Francisco, CA 94107



Whoever said Games were all about winning and losing... was right - so get Perforce's Fast Software Configuration Management on your team.

Perforce's Software Configuration Management System is the choice of winners, because no other tool can match the speed, control and reliability that it brings to the management of digital assets (binary and text files).

Maintain your top speed no matter how many users or files. At the core of Perforce lies a relational database with well-keyed tables, so simple operations can be accomplished in near-zero time. Larger operations (like labeling a release and branching) are translated into keyed data access, giving Perforce the scalability that big projects require.

Install it fast, learn it fast, execute operations fast. With other SCM systems, developers face an unpleasant choice: do it the right way or do it the fast way. Perforce's speed and reliability mean the fast way is always the right way.

Work anywhere. Perforce is efficient over high-latency networks such as WANs, the Internet and even low-speed dial-up connections. Requiring only TCP/IP, Perforce makes use of a well-tuned streaming message protocol for synchronising client workspace with server repository contents.

Develop and maintain multiple codelines. Perforce Inter-File Branching™ lets you merge new features and apply fixes between codelines. Smart metadata keeps track of your evolving projects even while they develop in parallel.

Supports all development platforms. Perforce runs on more than 50 operating systems, including all flavours of Windows, UNIX, Linux, Mac OS X and more.

Integrate with leading IDEs and defect trackers: Visual Studio.NET®, Visual C++®, Visual Basic®, JBuilder®, CodeWarrior®, TeamTrack®, Bugzilla™, ControlCenter®, DevTrack® packages, and more.

PERFORCE
SOFTWARE

Fast Software Configuration Management www.perforce.com

Download your free 2-user, non-expiring, full-featured copy now from www.perforce.com
Free (and friendly) technical support is on hand to answer any and all evaluation questions.

All trademarks used herein are either the trademarks or registered trademarks of their respective owners.

INDUSTRY WATCH

KEEPING AN EYE ON THE GAME BIZ | *everard strong*

Nintendo gets slapped with the European Commission's fourth-largest fine. Resulting from charges of price-fixing on the European continent, Nintendo has been told to dole out a record €149 million ($150 million) in fines. The fine was the fourth highest handed out by the commission. Nintendo officials are not contesting the charges but expressed surprise at the amount of the fine, claiming the commission should be more lenient, as Nintendo volunteered the information to the EC in the first place.

GRAND THEFT AUTO: VICE CITY breaks big numbers. Industry analysts predict that Rockstar Games could sell as many as 10 million copies of GTA: VICE CITY in 2003, generating more than $400 million in revenue, a number that puts it on par with major Hollywood blockbuster film releases. Epic/Sony is releasing seven separate CDs featuring the music played on each of the seven radio stations featured in the game.

Console market is growing up. New studies by Jupiter Research show that the median age of game players is 23, 61 percent of those surveyed favor "low-intensity puzzle and board games," and one in 10 online adults owns a current-generation console (Playstation 2, Xbox, or Gamecube). The overall audience of game players across all platforms will grow 18 percent, with a 61 percent penetration of the market by 2007. The number of game players who play more than 15 hours a week will grow by 40 percent, from 12 million to 17 million by 2007.

Rockstar Games' GRAND THEFT AUTO: VICE CITY is giving Hollywood a run for its money.

Time Warner rolls out G4. G4, the Comcast-owned videogame television network, signed a carriage agreement with Time Warner Cable that will see the network rolled out to all of Time Warner's digital basic subscribers over the next two years.

Sega's expectations fall. Sega has slashed its half-year net profit forecast by 77.8 percent, forecasting to sell 4.6 million units of game software by the end of this fiscal year ending March 2003, compared to earlier predictions of 6.2 million units. Sega cites in part poor sales of its NFL 2K3 title for this downturn.

Xbox's Ohura steps down. Hirohisa Ohura, Microsoft managing director for Japanese Xbox operations, is leaving the post on January 1 and moving to the company's U.S. headquarters. No reason was given for his departure at the time. Microsoft immediately began the search for a replacement.

ESRB names new president. The Entertainment Software Rating Board (ESRB) announced that Patricia Vance has been appointed its new president, replacing Dr. Arthur Pober, who led the ESRB from its inception and helped design the rating system used today. Pober will still serve as the ESRB's president emeritus.

Interplay to focus on content. Following its financial woes, Interplay has announced that a series of as-yet undisclosed moves will shift the company to a "content-driven" focus. Personnel shifts within the company have resulted in Philip G. Adam being named as the company's new president and Gary Dawson as chief operating officer.

THE TOOLBOX

DEVELOPMENT SOFTWARE, HARDWARE, AND OTHER STUFF

Nokia unveils game-focused handset. Nokia revealed their N-Gage, a combination mobile phone with game input controls and a full-color screen. Scheduled for a February 2003 release, Sega has signed on to develop games for the device. www.nokia.com

Dolby introduces new virtual speaker technology. Dolby Laboratories announced Dolby Virtual Speaker, a technology that reproduces the dynamics and surround sound effects of a 5.1-channel speaker system on a dual-speaker PC. www.dolby.com

Vicon ships Vicon IQ. Motion capture company Vicon introduced Vicon IQ, software that, by retaining data from each stage of the capture process, reduces inaccuracies and minimizes time-consuming manual editing. www.vicon.com

UPCOMING EVENTS CALENDAR

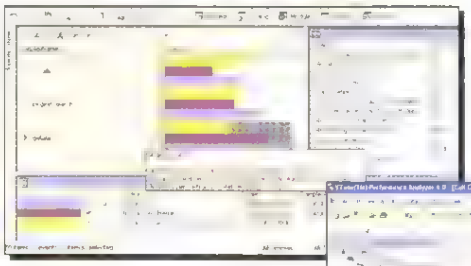
MILIA
PALAIS DES FESTIVALS
Cannes, France
February 4–6, 2003
Cost: variable
www.milia.com

GAME DEVELOPERS CONFERENCE
SAN JOSE CONVENTION CENTER
San Jose, California
March 4–8, 2003
Cost: $150–$1,975
www.gdconf.com

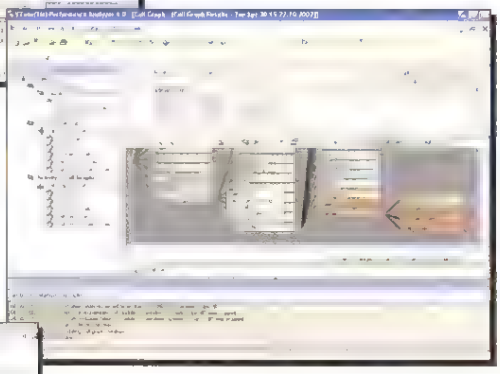
GDC MOBILE
SAN JOSE CONVENTION CENTER
San Jose, California
March 4–5, 2003
Cost: $895
www.gdcmobile.com

Maximize Your Application Performance with Intel® Software Development Products

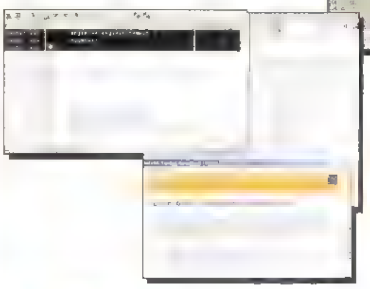
Intel, the world's leading processor vendor, provides professional software development tools. These tools are designed to integrate easily with the development environment you're already using. All tools include 1 year of free product upgrades and Intel Premier Support. Take advantage of the advanced features of Intel® Pentium® 4, Intel® Xeon™, and Intel® Itanium® processors with tools from the people who know processors best.



Time & Event Sampling
View performance by time or processor events (cache misses, branch mispredictions, etc.)



Call Graph Profiling
View call paths, the number of times a function is called, time spent in each function and more.



Intel Tuning Assistant
Uses Intel compiler technology to give advice on improving code performance.

Intel® VTune™ Performance Analyzer

VTune™ Performance Analyzer 6.1 helps you increase your software performance by enabling you to identify and analyze bottlenecks in your code. The VTune analyzer has many powerful analysis features that offer useful guidance on how to improve performance. Version 6.1 supports VS.NET*, C# and multi-threaded applications.

"The Intel VTune Performance Analyzer took a multi-day task and turned it into a sub-day task."

—Randy Camp
V.P. Software Research and Development
MUSICMATCH, Inc.

Intel® C++ and Fortran Compilers for Windows* and Linux*

Intel® has applied its expertise in software optimization for Intel® 32 bit and Intel® Itanium® processor-based systems to develop the next release of its compilers product-line: Version 7 of the Intel® Compilers. The Intel Compilers 7.0 deliver optimal performance on Pentium® 4 and Intel® Xeon™ processors, and the 64-bit Itanium and Itanium 2 processors and take advantage of Multi-processor systems and Hyper-Threading technology.

Intel® Performance Libraries

The Intel Performance Libraries simplify development of high-performance applications. These libraries free developers from having to write separate versions of code for different processors by providing a simple, consistent API interface that works across processor architectures. The libraries include highly optimized processor-specific code for graphics, multimedia, mathematical and signal processing functions.

Intel® VTune™ Performance Analyzer 6.1	I23 0170	699	$534.99
Intel® C++ Compiler 7.0 for Windows*	I23 0A10	399	$305.99
Intel® Math Kernel Library for Windows*	I23 0180	199	$173.99

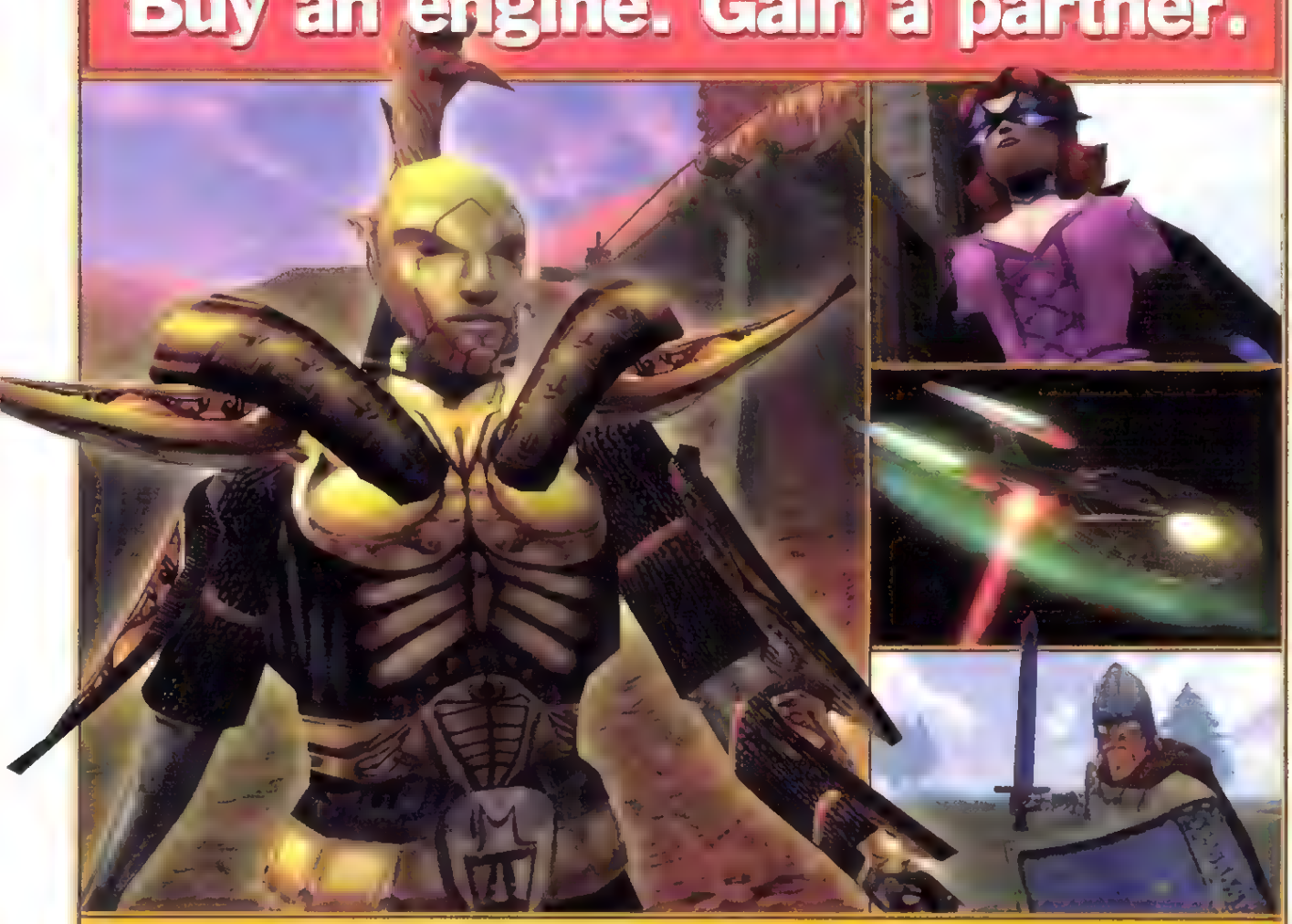
* *Also available for Linux*

FREE 30-day trials available at:
programmersparadise.com/intel/gdm

© 2002 Intel Corporation. Intel, the Intel logo, Pentium, Itanium, Intel Xeon and VTune are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries. *Other names and brands may be claimed as the property of others

03CT01C

Buy an engine. Gain a partner.



When you license NDL's NetImmerse 3D game engine and development toolkit, you get much more than great code. You gain a development partner with engineers dedicated to doing everything possible to ensure the success of your title.

NetImmerse is the only engine that can deliver all of this:

- A complete toolset that accelerates your art pipeline.
- Optimized performance across all platforms.
- Full source code for all libraries and tools.
- Responsive support from those who develop the engine.

Great software. Solid partner. Proven success.

NetImmerse
The fastest path from conception to completion

916.783.4380 • sales@ndl.com

GameDeveloper FRONTLINE FINALIST '02

ndl

The Elder Scrolls III: Morrowind courtesy of Bethesda Softworks
Star Trek: Bridge Commander™ courtesy Paramount Pictures. ™, ® & © 2002 Par. Pic. All Rights Reserved.
Dark Age of Camelot courtesy of Mythic Games.
Tennis Master Series courtesy of Microids.
Freedom Force courtesy of Irrational Games

www.ndl.com

Adobe's Photoshop 7.0

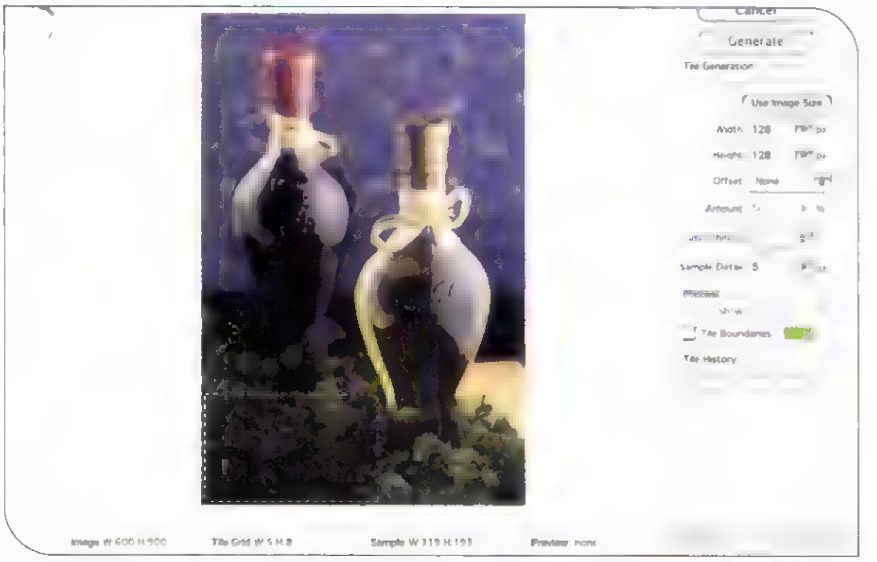
by spencer lindsay

Out of the box, the interface for Photoshop 7 looks very similar to the 6.0 update. However, upon further inspection, some really cool new features emerge that are especially useful in games production. Since Photoshop 7 comes with so many features in all, I'm going to focus solely on the new ones.

Healing Brush and Patch tools. The Healing Brush is similar to the Clone Stamp tool in previous versions, but with a twist: when you select a source area of your image to duplicate, it only clones the texture of the sample, not the luminance values or color information, which is great for creating seamless tiles for texture maps.

The Patch tool is for bigger areas and works somewhat the same way. Select the area you want patched and then drag the selection to the new texture area you want to clone from. Again, this won't clone the pixel chroma or luminance values, just the texture of the source. I played around with the tool for a while and found it easy to remove blemishes and random image noise from a picture.

Pattern Maker. The new Pattern Maker tool is excellent at creating tiling patterns from a completely unusable source; you select a section of the source image and the Pattern Maker creates an endless number of randomly patterned, tiling textures. Like most features in Photoshop, it's very straightforward and easy to use.



Photoshop 7's Pattern Maker tool holds great potential for 2D texture artists.

File browser/web gallery. There are some great features in the file browser, specifically the ability to save the Exchangeable Image Format (EXIF) information data from your digital camera with each image. All the settings from your digital camera — the date, the time, and whether the flash was fired during the shot — is saved in the EXIF window, and it keeps that information with the image after saving it.

Another useful feature of the file browser is batch renaming. If you've got a whole load of images that you took at E3 and they're all named "P000098234.jpg," you can rename them all to something more useable and transfer them to another directory. Using the web gallery feature, you can place them in a web page with icons and navigation buttons and upload them in a matter of seconds. The only beef I have with this feature is that it makes you save your pages with the .HTM extension instead of the more traditional .HTML and you have to trim the `<meta name>` tags out of each file.

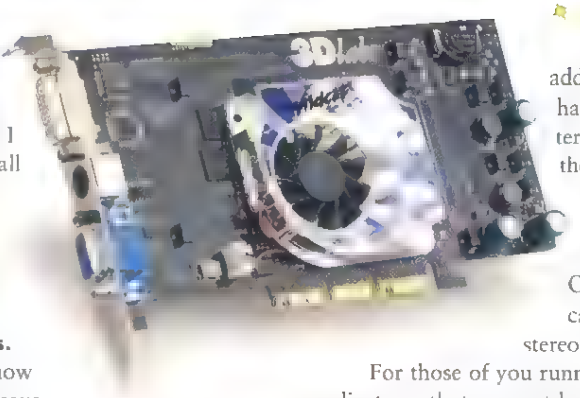
New brush features. The brushes are now modifiable in so many ways that you can simulate painting with pretty much anything. You want to put oil paint on canvas with a camel hair brush?

SPENCER LINDSAY | *Spencer is president at Etribe Studio in Santa Cruz, Calif., a real time 3D and multimedia content provider for the game industry. He spent five years at Atari Games as art director, where his titles included* San Francisco Rush, San Francisco Rush: The Rock, Metal Maniax, *and the design phase of* San Francisco Rush 2049

How boring. How about axle grease onto rubber with a dead fish instead? Done. Blood on stone with a hand? Done. The scatter function combined with texture allows a wide variety of realistic, smudgy, natural effects that were much more difficult to achieve in the older versions. I needed a "ground texture" for our game the other day and in 10 minutes was able to crank out some realistic-looking grass and dirt that tiled nicely.

In addition to a slew of parametric controls, you can set the brushes to work with Wacom's pen angle, airbrush wheel, and a passel of other customizable pen controls.

Liquify. The enhanced Liquify command is great for subtly moving portions of a texture map to fit a mesh. When I build characters, I always find that one spot that doesn't match up exactly with the UVs on my model. Using Liquify, I can nudge small or large areas of the texture map into the right alignment.



Tool palettes. Photoshop 7 now allows you to save tool palettes and brushes in a user-configured preset that can open automatically. In addition, you can save "Tool Presets," yet another palette in the History and Actions tab. In the new version, if you come up with a tool setting for your paintbrush that you like, you can set that as a preset and then later on access that with a button click.

Bottom Line. Photoshop 7 is an excellent upgrade and has enough new features and tools to warrant purchase. The new brush dynamics and the Healing Brush tools alone make the upgrade worth it. So go out and get a copy and then spend the rest of your life in a dark room, staring at a CRT.

STATS

ADOBE
San Jose, Calif.
(408) 536-6000
www.adobe.com

PRICE
$609 (MSRP)

SYSTEM REQUIREMENTS
Mac: G3 or G4, Mac OS 9.1–10.1.3; 128MB RAM (192MB recommended); 320MB disk space; 800×600 color monitor with 16-bit color or greater video card.
Windows: Pentium III or 4, Windows 98, ME, NT, 2000, or XP; 128MB RAM; 280MB disk space; 800×600 color monitor with 16-bit color or greater video card.

PROS
1. Great new brush functions.
2. File browser and batching options.
3. More specific tools make creating tiled textures easier

CONS
1. Lack of hotkey customization.
2. Still Mac-centric at its core.
3. Emphasis on web-specific features instead of traditional ones.

3Dlabs' Wildcat VP

by ron fosner

The Wildcat VP is built around 3Dlabs' new P10 chip, a real-time multithreaded processor that allows you to run single or multiple accelerated graphics applications with increased performance. The Wildcat VP cards can address up to 16GB of virtual memory, with the cards having 64MB or 128MB of on-board memory. Therefore, these cards are not designed to improve performance on older machines, they are designed for machines with Pentium 4 or Xenon processors with lots and lots of RAM running Windows 2000 or XP. They will run on Windows 98 or ME, but it's not recommended.

The sheer beefiness of these cards is impressive. Not only can they run accelerated OpenGL in a dual monitor setup, but each monitor is capable of running in 2048×2048 resolution in 32-bit color. In addition, if you have LCD shutter glasses, there's a stereo output allowing you to run an OpenGL application in stereo.

For those of you running modeling applications that support lights, the Wildcat VP line supports 24 hardware-accelerated lights. The Acuity driver interface lets you select performance options for either high texture requirements (as in a modeling application) or lots of geometry.

There are four Wildcat VP models, from the new low-end VP560 ($249) to the ultra-high end VP870 ($1,199), with the differences being in speed and memory configurations.

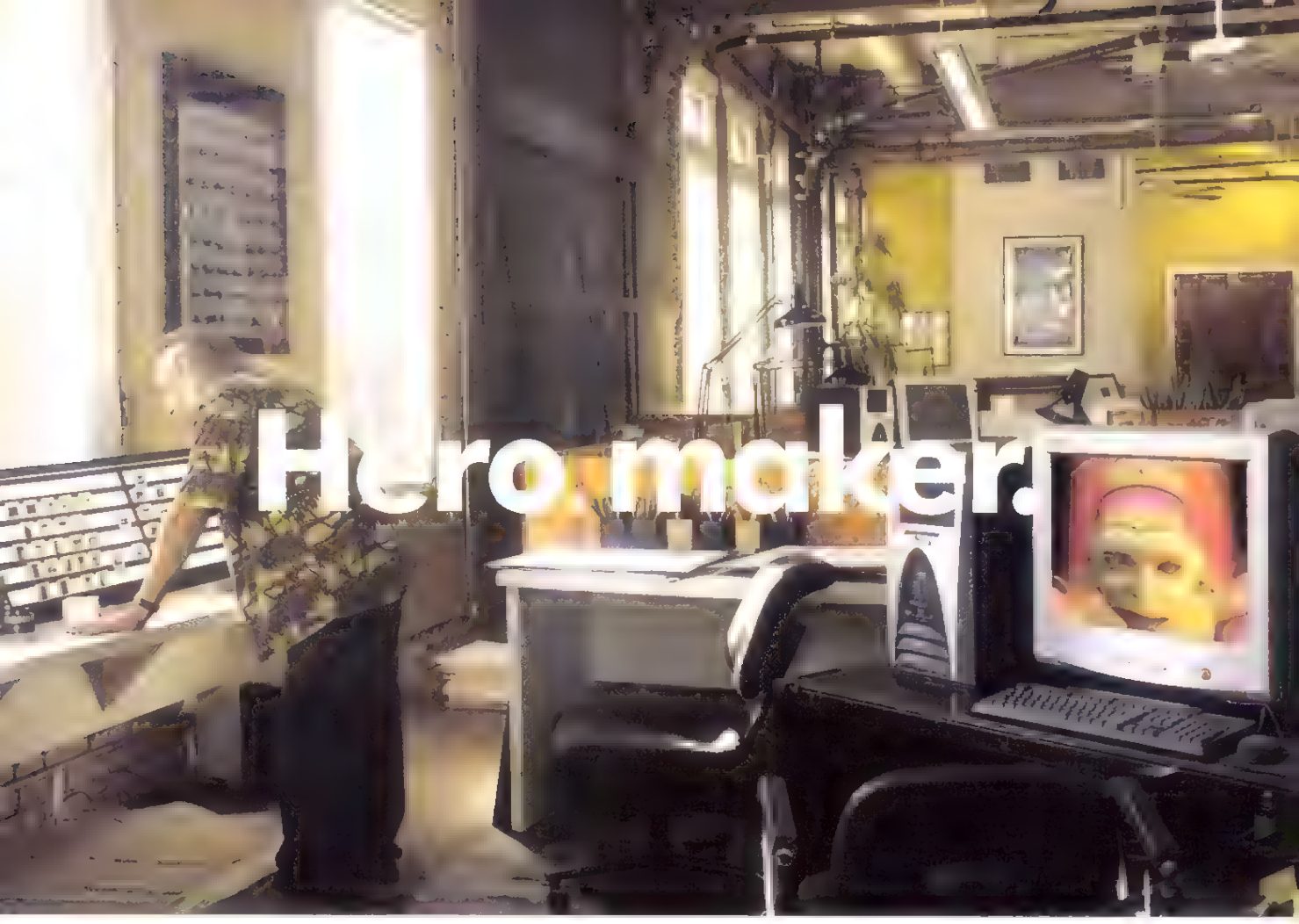
These cards are fully DirectX 8.1–compliant, and they will have DirectX 9 drivers. The major DirectX 9 features that the Wildcat VP doesn't have are 2.0 pixel shaders (they do support 2.0 vertex shaders) and floating-point pixels. This means that there's no hardware support for the extended precision color operations that require 64- or 128-bits-per-pixel color space.

OpenGL support, as you might guess, is excellent. The most interesting aspect is that 3Dlabs and ATI are the driving forces behind the OpenGL 2.0 specification, and 3Dlabs has a set of prerelease drivers available. All of the specifications can be found at www.3dlabs.com/support/developer/ogl2/index.htm.

If your day-to-day activities include working on modeling in 3DS Max or Maya, or you're looking for a high-end graphics card for that über-work PC, or you're looking to try out the latest OpenGL shader language, then the Wildcat VP deserves serious consideration.

★ ★ ★ ★ | WildcatVP
3Dlabs | www.3dlabs.com

Ron Fosner is a 3D programmer and consultant. Reach him at ron@directx.com.



Your creative awards called. Said they're waiting to happen on the Evo Workstation W8000, from the new HP. Designed and optimized for intensive graphics and the latest video applications, this HP Evo Workstation with an Intel® Xeon™ Processor turns creative juices into rivers. This, plus our legendary software partnerships, now means that designers, animators, editors and developers have even more reason to get all giddy. Mix with way-out-there ideas and clear some shelf space for those awards.

Ready to strut your stuff?
Visit www.hp.com/go/evoworkstations
or call 1-800-888-0414

The Power of
MACROPROCESSING

invent

HP PCs use genuine Microsoft® Windows®
www.microsoft.com/piracy/howtotell

©2002 Hewlett Packard Company. During the HP/Compaq product transitions, some HP workstation products and packaging may be labeled with the Compaq brand. Intel, the Intel Inside logo and Xeon are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries. Screen image courtesy of Softimage Co. and Avid Technology Inc. Softimage/XSI is a trademark or registered trademark of Softimage Co. and Avid Technology Inc.

American McGee: Silver Screens & Pixellated Dreams

There is a belief in the game development community that game teams can, and should, be assembled the way they are around films: gather individuals and groups for each project, without committing to long-term contracts or employment. American McGee, with his experiences in game development (id, Electronic Arts, and his own development company, Carbon 6), and in other media projects, including film and video, shared his views on movie making, game making, and, in his opinion, how separate the two worlds really are.

Carbon 6's American McGee ponders what Hollywood and game developers can teach each other.

Game Developer: Do you think the game industry adopting Hollywood's methods of forming teams on an as-needed basis is a viable one?

American McGee: Bringing teams together on an as-needed basis only works in an industry that has standardized tools, language, and practices. This works well in the film industry, as they've been working with the same tools for several decades.

On the other hand, games are almost always started from scratch with new lighting systems, animation systems, a scripting language, and often with basic game-mechanic principles being re-invented. The in-house vocabulary related to the tools and game design is also being invented on the fly. This necessitates having key members of a given development team stick together from project to project. If you disband, you lose the tribal knowledge of what was built last time around; we're not recording that history right now.

GD: Physical proximity is key to Hollywood keeping their production system workable. Does the game industry's lack of a Hollywood and Vine impede the incorporation of such a system?

AM: There is a "Hollywood and Vine" for the games industry: the Internet. One of the projects I'm currently managing is being created by individuals in Australia, Los Angeles, and Texas. It's a global development team in a very real sense, and it's working marvelously. It's allowing me to use the very best resources from around the world.

The only thing I've ever seen come out of developers who are near one another is tons of employee theft, gossip, caveman rivalry, and bitterness. But then, maybe that was just something weird in Dallas.

GD: How has the film industry's progress differed from game development's so far?

AM: We went from 2D black and white [games] to 3D faster than film went from black and white to color. And we're still going. Eventually, you and I will plug our computers into our heads and escape into any world we want, surrounding ourselves with a completely new reality. The interface is coming, the content is evolving, and whether they know it or not, game creators are on their way to being alternate reality creators.

GD: What do you mean by alternate reality creators?

AM: PONG is a game. DEFENDER is a game. PAC-MAN is a game. DOOM and QUAKE, they're crossing over from game into alternative reality. GRAND THEFT AUTO 3, an extremely important game for our industry, goes much further away from game. It's an escape, but at the core of it, it's not even a game. It's a narrative, and you are part of the story, and it's a sandbox in which you can go play around. A game is almost a misnomer for what we're eventually turning ourselves into as an industry. Do you call a book a game? Do you call a piece of fiction a game? No. Those things are attractive to people because they're an escape, and whether you've escaped through the page, or television, or film, or music, it's about letting people have an alternative life, an alternate reality. That's what I mean by alternate reality creators.

GD: Do developers limit themselves by thinking in terms of creating games instead of experiences?

AM: I think as an industry, we don't really think about just how big of an impact we can have with our medium. The word "game" makes it seem small to a lot of people. The collective subconscious definition of a "game" is still stuck in the 1980s with some kid pumping quarters into an arcade game.

In Hollywood, a lot of people I've talked with view games as an inferior medium, and they have a right to because a couple of years ago they really were. But now, as people here recognize what can be done, they are starting to freak out partly because they're seeing a lot of their sales cannibalized by those of games. You meet with them and they're like, "We don't want to get involved with games because it will just help games grow bigger." That's backwards thinking. Games are going to keep growing, so you better jump on or get lost.

GD: Do you think we're at the stage where people are going to start test-marketing titles as they do movies? Will there be a panel of mall-goers giving their input on the next GRAND THEFT AUTO?

AM: The great thing about GRAND THEFT AUTO is that number three in the line was the big hit. People forget that there where two of them. It was something that wasn't right the first time, and it wasn't quite right the second time either. But the third time around, man, they got it. Now they've got a formula for success, and they shouldn't screw it up by trying to add the kitchen sink to it, which a lot of game people tend to do.

NOTHING BUT MEAT!

MICROSOFT DIRECTX DEVELOPER DAY WEDNESDAY, MARCH 5, 2003

Register for Microsoft® DirectX® Developer Day Sponsored Tutorial during the Game Developers Conference and learn about prime technologies like:

- **High-Level Shader Language**
- **Nontraditional Lighting and Techniques**
- **Ultimate Rendering Realism**
  - **Spherical Harmonics**
  - **High Dynamic Range Techniques**
  - **Latest Innovations from Graphics Research**
- **Tools and Technologies to Make Writing Games for Windows® Easier and Better**
- **Power Debugging with Microsoft Visual C++® and the DirectX Extensions**
- **Interactive Analysis of Windows-based Games with D3DSpy**

MICROSOFT SPONSORED SESSIONS THURSDAY, MARCH 6, 2003

Get a taste of these beefy Sponsored Sessions that pack a lot of meat:

- **Opportunities in the PC Gaming Marketplace**
- **Managed DirectX: Microsoft .NET support for DirectX 9**
- **Screaming at the Machine: Using Speech Recognition as a Complement to Traditional Game Input Techniques**
- **Best Practices for DirectX 9 Audio Programming**
- **How to Write a Network Layer**

Sponsored By: Microsoft

Best When Used On: 3/5/03 & 3/6/03

Register online for the Microsoft DirectX Developer Day and the Sponsored Sessions at:

www.gdconf.com

Get the latest on DirectX at **http://msdn.microsoft.com/directx/**

Microsoft® Windows® Gaming

© 2003 Microsoft Corporation. All rights reserved.
Microsoft, DirectX, Visual C++, Windows, and the Windows logo are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries. The names of actual companies and products mentioned herein may be the trademarks of their respective owners.

Microsoft®

Interactive Profiling, Part 2: Behavior Analysis

Last month ("Interactive Profiling, Part 1," December 2002), I talked about the sorts of low-production-value profilers we often need to build into our games. I described these as "interactive profilers," distinguishing them from the commercial products available, which I called "batch profilers."

I also described several ways that an interactive profiler can be helpful when a batch profiler cannot. A big problem with batch profilers is that they average measurements across the profiling session. If the program changes its overall behavior several times during the session, a batch profiler offers some kind of average behavior which may not be much like the specific behaviors you want to debug.

To illustrate this point, Table 1 shows an example of three different behaviors that a program might exhibit and the resulting analysis you'd get from a program such as VTune. During any given frame, the readout of an interactive profiler will clearly show one of these three behaviors, each of which has a single dominant function eating most of the CPU time. The result given by the batch profiler shows no such dominance.

Now suppose that behavior C is actually an error, and it should not be happening at all. Given a clear picture of these numbers, you see that there is a dominant function in behavior C and we have a hope of understanding why that function is called. The result of the batch profiler shows three functions of roughly equal importance, and it's unclear how to proceed in optimizing the system. In practice, modern games are even less clear than this, since they involve much longer lists of profiling numbers and a broader gamut of behavior.

An interactive profiler helps clarify problems such as these.

TABLE 1

Behavior:	A	B	C	Batch Average
rendering	75%	25%	25%	42%
physics_sim	15%	65%	15%	32%
ai_pathfind	10%	10%	60%	27%

But historically, interactive profilers have provided minimal functionality: a list of program regions and CPU times and not much else. The numbers they present to us are ephemeral — either they exist for a frame and then disappear, or they're averaged over several frames, hiding spikes that are important to detect. This is an interesting conundrum in interactive profiling: we want to add hysteresis to the reported data so that it is easier to read (as with the IIR filters I discussed last month), but that hysteresis can hide important events. Hysteresis makes the profiler less interactive and more batch-oriented, but without the stored-data advantages of batch profiling.

So with the common kind of interactive profiler, we lose one of the strengths of batch profiling: the capability for precise and unhurried analysis of the available numbers. We can see changing behaviors in the live profiling data, but they're difficult to quantify. How much time is being spent in behavior A and how much in behavior B? How frequently is some particular performance spike occurring, and how long does the spike last?

In a sense, we want to quantify the behavior of the profiling numbers at a meta level. The raw list of numbers for any particular frame tells us how long the program spent drawing the world, running AI routines, performing collision detection, and so on. As we saw with Table 1, those numbers can change significantly during a run. Really we want to step up a level and not limit ourselves to speaking only about elements of the raw timing data. We want to look at the array of data from any given frame and treat it as a coherent set of numbers, as an abstract object in its own right.

That coherent set of numbers is a way to quantify what I have been calling a "behavior." We want to talk about how consistently the game produces that behavior, and, when it produces some other behavior, precisely what that means. We want solid, quantitative information about when and how often the program exhibits each behavior.

Thus we can elevate ourselves to a level where we are taking concrete measurements about formerly abstract ideas such

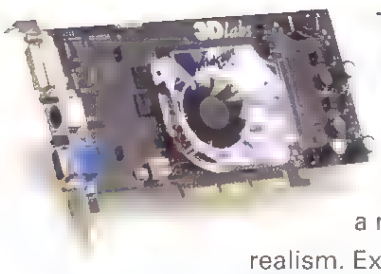
JONATHAN BLOW | *Jonathan (jon@number-none.com) is a game technology consultant. Right now he is hanging out in Austin and writing the "Bio" section for his column.*

A New Design Freedom

3Dlabs
Graphics Evolved
A CREATIVE Company

GameDeveloper FRONTLINE FINALIST '02

Introducing the World's First Fully Programmable Visual Processor



The sky's the only limit to your design creativity with Wildcat® VP—an astounding breakthrough in 3D workstation graphics. The industry's first fully programmable Visual Processing Unit (VPU) powers a family of accelerators that deliver unmatched productivity for professional OpenGL® and Direct3D®—based applications. But that's not all. With unrivaled flexibility, Wildcat VP is enabling a new generation of applications to extend the boundaries of interactive visual realism. Experience the ultimate in graphics design freedom today. Be ready for visual programmability tomorrow. Wildcat VP from 3Dlabs.

Leading OpenGL and Direct3D performance	A full family of 64MB and 128MB boards
Over 200GFlop and 1.2 TeraOp VPU	High performance virtual memory
Professional-grade reliability and quality	Dual independent displays

Wildcat VP

www.3dlabs.com

3D abs is a registered trademark of 3Dlabs Inc. in the United States and other countries

as "the fill-rate-limited behavior" and "the physics-heavy behavior." We will no longer be trying to make inferences by looking only at timing data for individual program zones. If this elevated data is presented in an easy-to-perceive way, it will become an important tool for understanding just what our program does.

This month, I want to explore the kinds of analysis capabilities we might want in an interactive profiler following up next month with some concrete implementation. My goal will be to introduce as much analysis as possible without slowing down the running game, which is important for reasons I discussed last month.

What Do We Want?

To design appropriate analysis tools, we first need to define clearly what we want the tools to do. This clarification may take some thinking; as games have evolved, our profiling needs have also changed.

First of all, instruction-level profiling is no longer as important as it once was. For one thing, modern CPUs are unpredictable, and it's harder than ever to make a program faster by applying minor code tweaks. But there's also a deeper reason.

It's a piece of ancient programmer wisdom that we should not resort to peephole optimization (which is what instruction-level profiling is good for) until the options for algorithmic optimization have been carefully explored. High-level changes to an algorithm can functionally invalidate low-level optimizations, while the overall impact on the execution time dwarfs the gain of those low-level tweaks.

As time goes on, algorithms used in games are getting more complicated. Thus, the number of degrees of freedom for possible algorithmic changes is constantly increasing. In other words, we have more choices of different ways to alter an algorithm while still accomplishing the same goal. With so many degrees of freedom and with limited manpower to spend on optimization, we can easily exhaust our work budget exploring algorithmic changes without reaching the low-level optimization stage for most of our game's subsystems. Thus, we want a framework that encourages and supports profiling at an algorithmic level, not an instruction level.

Traditional profilers also assume that a program's behavior is somewhat orderly and that a static listing of CPU times is sufficient to understand its behavior. But games are so complicated now that we often don't have a good idea of what they're doing internally, even when they seem to work.

We want a framework that encourages and supports profiling at an algorithmic level, not an instruction level.

For example, it's easy to accidentally draw all the entities in the world twice per frame. Big blocks of code may have gotten moved around, but a function call got duplicated. In this case the entity-rendering routines get called twice, but it's hard to see because the calls are in different files, a few function calls' depth apart from each other. If you're using multipass rendering, there will already be multiple calls to some entity-drawing routines that you actually want to be there, further confusing the situation.

A more sophisticated way you might render duplicate entities is to collect them by portal navigation through the world, but in this case the ones you have already collected don't get marked correctly. Odd combinations of portals would then cause entities to be added to the render list multiple times.

Another example of seemingly correct behavior concealing incorrect behavior might arise when performing collision detection against an excessive number of faraway objects. Either bounding volumes for the objects are unintentionally large, or the data structure they're being stored in is overly loose. In both this and the duplicate-drawing case, the rendered scene appears to be correct, but the game has taken a lot of extra time to generate it.

Thus, one of the main purposes of a modern profiler should be to act as a tool to understand the behavior of a program, to verify that the program is running correctly or to understand how it is wrong. Even when performance problems are not due to bugs, they often exist because we don't quite understand the performance ramifications of our algorithms when given complex input data. The chief job of the profiler is to provide us with such an understanding.

Effectively using such profiling data requires an approach that is a little less direct than what we had in the past. With instruction-level optimization, the profiler can point to the exact piece of code with which we should be concerned. But when optimizing algorithmically, the code where the time is being spent is not usually what we care about. We actually want to know why time is being spent there so we can change the program's overall behavior, by either calling that code less or eliminating it entirely.

Imagine rendering stencil shadow volumes. The profiler tells us we're spending a lot of time computing the silhouettes of the objects in our game world, so that the shadow planes can be extruded from

Did a bloated .ELF crash your holiday party?

"CodeWarrior's short compile and link times during development have improved our ability to stay on schedule and ship on time!"
Craig Broadbooks, Solworks

Start the New Year off right! CodeWarrior™ development tools from Metrowerks shrink your bloated .ELF and won't weigh down game development.

Did a big, fat .ELF block your games from getting out the door last year? Make a resolution to get your execution files back down to a normal size, so you can ship on time next year. You can do it with a CodeWarrior™ compiler — without forcing yourself to trash that sweet game code. Save even more time with CodeWarrior analysis tools that pinpoint and solve additional performance bottlenecks. And this year, lose the bloated .ELF!

CodeWarrior
Game Development Tools

Try an evaluation copy and ship on time next Christmas!
Request an eval at www.metrowerks.com/games or
call +1 512-997-4701, within the U.S. call 1-800-377-5416

NINTENDO GAMECUBE
and Game Boy® Advance
tools and middleware vendor

Tools & Middleware

© 2003, Metrowerks, Corp. All rights reserved. Metrowerks, the Metrowerks logo and CodeWarrior are registered trademarks of Metrowerks, Corp. in the U.S. and other countries. NINTENDO GAMECUBE and Game Boy are registered trademarks of Nintendo Co., Ltd. PlayStation is a registered trademark of Sony Inc. All other trademarks remain the property of their respective owners. MG91141A

metrowerks
Software Starts Here

them. Once we are convinced that the silhouette computation is not grossly incompetent, we look for algorithmic solutions to the problem, which will likely yield much greater speed-ups than trying to optimize the silhouette computation itself. One such speed-up is to cull more aggressively the light sources that only create shadow volumes that cannot be seen. With fewer light-to-object interactions, we call the silhouette routine less often, and the game goes faster.

While understanding the behavior of our program is a very important goal, right now we have no tools for that purpose. That seems like a bad thing.

How We Might Use Such a Profiler

So what specifically do we want the tools to do for us when we sit down in front of the computer to profile? First of all, I'd like to be able to casually run the game and get profiling data, without needing to prepare a special build and without a large frame-rate hit. This ability allows me simply to walk around in the game world and do things that test performance characteristics.

In addition, I'd like the profiler to detect the various ways that the program behaves and to provide me with an abstract list of those behaviors with quantitative results attached. After I run around the game world doing a bunch of stuff, I want the profiler to give me something like Table 1 and say, "You were in behavior A 50 percent of the time, B 35 percent of the time, and C 15 percent of the time." A, B, and C each consist of a separate set of profiling data, like a batch profiler would give us for an entire run. Once I'm viewing things from a higher level, I would like to know how important each behavior is and how the behaviors interact. It would be nice to see a graph of which behavior was happening when, so I can correlate the changes with things that happened in-game.

Once a well-defined set of behaviors has been measured, I'd like the facility to mark some of them as expected/acceptable behaviors and some of them as anomalous/unacceptable behaviors. The profiler can provide statistics about how much time was spent in unacceptable behaviors. Suppose the game has a journaling system, where I run a recorded session several times to benchmark performance as I attempt several optimizations, to see how effective those optimizations are. I can watch the percentage of time that is spent in unacceptable behaviors and the CPU measurements for those behaviors in isolation. These observations may provide much more concrete and useful information than a batch-profiler-style average.

If the game has a journaling system, I'd like to mark certain behaviors as things I want to debug. Then I can start playing back the journaled game events; when the game reaches one of the target behaviors, the playback pauses. This pause allows me to access the problematic game states quickly and then look at the game world to infer what might be causing the problem.

In the previous stencil shadow example, there could be a behavior when the silhouette-finding takes unreasonable amounts of the CPU, but there doesn't appear to be an unusual number of light sources or objects in the scene. I could turn on wireframe rendering and see that far in the distance, in areas that are occluded from view, there are many light sources and objects. I can then deduce that there is an error in the occlusion-culling code. I could also find this kind of problem without a journaling system or analytical set of behaviors, by walking around the world and watching the live numbers. But the idea is that the extra analytical capability helps me find problems more quickly and in a more reproducible way, and also helps detect problems that are subtler.

I'd also like to have some querying ability about the behaviors. I want to be able to ask things like, "Which behaviors contain a lot of silhouette computation?" and then look at those behaviors as a group to understand them better. Perhaps I want to perform queries with Boolean operations, such as, "Which behaviors contain a lot of `collision_detect` and not a lot of `physics_simulate`?" This would indicate unusual situations where objects collided a lot without moving far between collisions, which is a behavior case I might want to spend a lot of effort to alleviate.

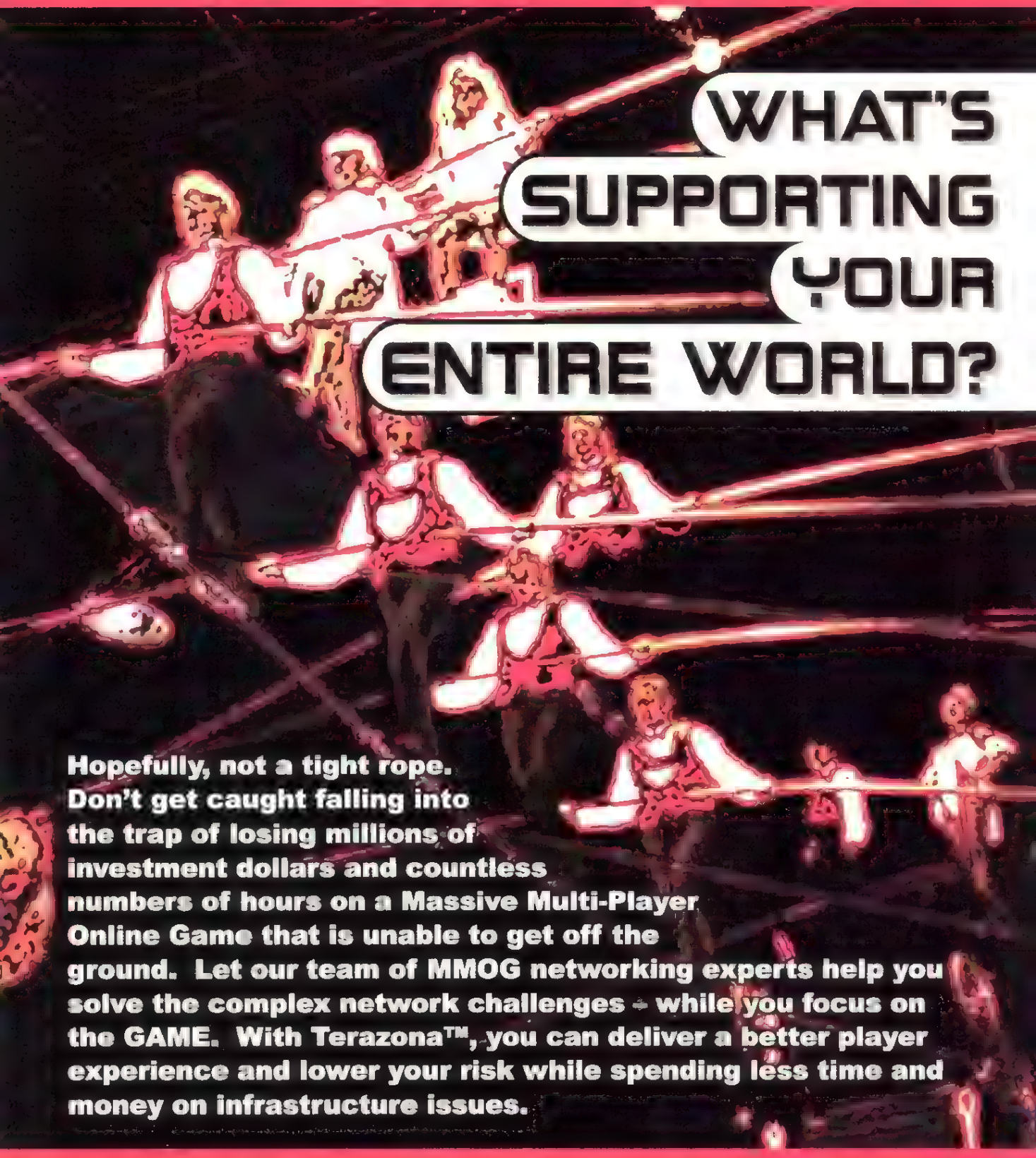
If I am investigating a certain behavior, I may wish to ask, "Which behaviors are like this one?" in order to find other cases that may shed light on the current situation. Or I may want to ask, "Which behaviors are very much unlike this one?" in order to see how far the game's behavior can deviate from what I am studying.

To discover performance patterns, I want the profiler to detect correlations between timing values. Suppose time spent in the dynamic texture generator is roughly proportional to time spent drawing particle systems, and both of those correlate significantly with increased frame time. That would indicate that the artists are using too many procedurally generated textures in the particle systems, and they need to cut down on those. But perhaps the other dynamically generated textures do not represent a performance problem, say, the ones used for rippling water.

Since the profiler is built into the game, I can expose game variables to the profiler. So instead of just classifying behaviors based on timing data, it can also look at "the number of monsters currently running around" or "the number of nodes in the potentially visible set." It can find correlations between these variables and the timing data, or perhaps just among the variables themselves. With that capability, I have a powerful tool tremendously different from anything I've had before.

Next Month

This month's column has consisted basically of a big wish list. Next month, I'll look concretely at the implementation of an interactive profiler that attempts to satisfy these wishes.



Hopefully, not a tight rope. Don't get caught falling into the trap of losing millions of investment dollars and countless numbers of hours on a Massive Multi-Player Online Game that is unable to get off the ground. Let our team of MMOG networking experts help you solve the complex network challenges - while you focus on the GAME. With Terazona™, you can deliver a better player experience and lower your risk while spending less time and money on infrastructure issues.

THE FUTURE OF MMOG DEVELOPMENT IS IN YOUR HANDS...

Zona,Inc.
www.terazona.com
Phone: 650.361.1133
E-mail to: sales@zona.net

TERAZONA

What's Wrong with Our Games? Part 1

Twenty or so years ago, in rooms filled with the tattered remnants of Christmas wrapping paper, families around the globe huddled in front of their TV screens as they fired up the very latest in interactive video entertainment. A new age had dawned, and the Frisbee and the toy soldier found themselves a nice, warm spot in the cupboard. Videogames had arrived.

The fuzzy white blocks were visually uninspiring on screens, but in the context of the technology of the time, we may as well have been plugging ourselves directly into the matrix. Entertainment had evolved into its newest form, and as it was forging a path through uncharted territory, every new gaming experience brought its own satisfaction, regardless of the poor visuals on display.

The fact that the one-man development studio maintained its presence in the game industry well into the 1980s shows that for many years players had relatively low expectations for graphics. But following the exponential curve of technological advancement, players came to expect a standard of visuals that do everything short of reaching out of the screen and slapping them in the face.

Today, like it or not, we have the joy of shooting at a constantly shifting target that moves away from us faster than our bullets can travel. The developer's job is increasingly about delivering quality, and as budgets continue to rise, taking production values with them, more is expected from us across the board.

Given that our job description now goes something like "make everything very nice, as quickly as possible," where as an industry are we falling short? Where could we improve?

To answer these questions, I recently began surveying various industry artists to see what their personal gripes were about the state of game graphics at present. This column is a consolidation of their impressions, coupled with my own, and thus reflects some of the range of opinions artists hold about where we are going wrong with our games.

Design Concepts

When questioned about the overall visual design of games, game artists identified the following areas of weakness:

Dystopia a-go-go. Once again, I find myself pointing the finger of blame at Ridley Scott (director of the movie *Blade Runner*). What is it with games and grimy, dirty worlds of filth? For that matter, what is it with science fiction's obsession with dirt? I think we can all agree that George Lucas did the right thing when he had squads of artists "dirty down" the locations and artifacts of the *Star Wars* universe, as the *Millennium Falcon* would have looked less authentic as a giant, squeaky-clean, plastic molding. But we must draw a line somewhere.

For videogames, creating a believable environment that is run-down and corroded is easier (and perhaps more enjoyable) than building someplace very clean and bright. But this principle has led us to a gaming canon packed with worlds that unfold as a thousand variations on the same theme. Surely there is a place for more visual originality somewhere in between the rainbow cuteness of Mario and *Blade Runner*.

Same worlds, same conditions. While we set too many games in dystopian future worlds, it is also true that we are guilty of following some other design stereotypes, not least of which is the path that leads us to the Slippy Slidey Ice World.

The low-friction game mechanic has popped up again and again almost since the dawn of game creation, with ice worlds used to increase the variety of gameplay. I'm not suggesting that this approach should be abandoned all together, but we should present it in a more original fashion. Visually distinguishing locations by using the desert world, jungle world, ice world, and volcano world is now most certainly stale and unimaginative.

Inconsistency. A game's overall design must strive to present the player with a consistent, stylistically focused game. Any art director or lead artist must spend a lot of time making sure that every artist on the team is working toward the same design standard. Still, problems are bound to occur when you're dealing with a collection of individuals.

HAYDEN DUVALL | *Hayden started work in 1987, creating airbrushed artwork for the games industry. Over the next eight years, Hayden continued as a freelance artist and lectured in psychology at Perth College in Scotland. Hayden now lives in Bristol, England, with his wife, Leah, and their four children, where he is lead artist at Confounding Factor.*

Illustration by Dominic Bugatto

You'll inevitably encounter the problem of differing levels of artistic ability. What can you do about this? First of all, the quality and abundance of the concept art can reduce the effect of individual discrepancies within a team. Sometimes it's necessary to rein in creative freedom to bolster consistency.

Task division can also be useful. If you handle textures and geometry separately, you can mitigate creative differences between artists somewhat. This solution may prove less rewarding for individual artists, but it is one option.

As a last resort, you may need to fire people who just aren't up to the standard you require. Reluctance to fire people is completely understandable, but when there is nowhere left to turn, it cannot be ignored as a solution.

If your time and budget allow, it is usually more reliable to have a smaller group working on a specific task for a longer time than a larger group covering the same ground quickly. Ultimately, the two alternatives cost the same in man-hours, but deadlines being what they are, you may not always have the luxury of choosing which approach to take.

In general terms, the better the artist, the more adaptable he or she is in terms of style. Ideally, a group of competent artists, focused with the necessary concept art and art direction, should be able to create a game world that holds together as a single entity.

Animation

Animation, almost more than any other area of game art, is the easiest to get wrong. Our perception of movement is extremely fine-tuned. Even though we are not often aware of it, we can recognize extremely small, seemingly invisible movement traits as being wrong or somehow unexpected.

As an example, the human walk cycle is a fairly standard range of movements, but with even the smallest changes in balance and gait, thousands of unique walks emerge. Think of how even from a great distance you can recognize the walk of a person familiar to you, even though their movement is not always perceptibly different from those around them.

Understandably, animation problems crop up in games on a regular basis. Among the most common are:

Muppet mouth movement. A personal favorite of mine is the character animation during speech where the head remains basically rigid, while the lower jaw flaps around in a sock-puppet imitation of conversation. Now before you get on your high horse and complain that that problem is entirely due to time restrictions, let me say that I know that this is usually the case, and that the same reasoning can be used to explain away most of the problems I'm highlighting. But my point in drawing attention to some of these offenses is to illustrate how, as expectations continue to rise, this poor level of execution will become unacceptable, and we'll need to address the problem if we wish to be successful.

So, what's the big deal about Muppet-mouth animation? Besides just looking stupid, it wastes an opportunity to open the game to the player in terms of personality and characterization. In any story-driven game, the interaction with other characters has the potential to provide one of the most rewarding experiences for the game player. Having wooden characters lowers each dialogue session to a dull period of "wait for the important information and then move on." Cutscenes present a whole range of problems besides that of character animation, but conveying emotion and personality during dialogue will increase the player's immersion in the game and thus reduce their desire to simply skip the talking and get back to the action.

So how do you make your characters more interesting? While there are some automated solutions out there, none seems to provide a magical fix. The best thing you can do is prioritize animators' time differently or find the resources to add more manpower to the task.

Move me like you should. Movement systems for many games seem to fulfill the minimum requirement for getting the player from one area to another without walking through walls. The days of hovering feet have regrettably not yet passed, where a character turning to the left or right seems to pivot around its center point while retaining the standing pose. Even more idiotic, we still see the wall-facing moonwalk that some characters do when they come up against an impassable object and contin-

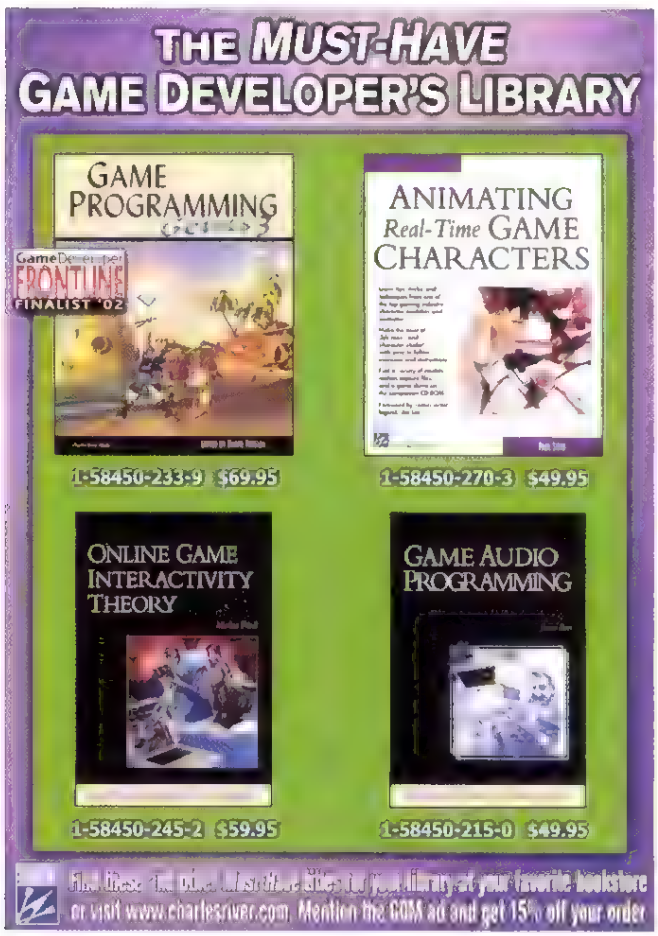
ue to go through their walk cycle without actually moving anywhere.

In both of these cases, a solution requires little additional work. Still, even some big-budget game developers have decided that these areas are of little importance, while at the same time they are happy to allocate resources to a team that has been asked to provide a realistic fluff-generation system for the carpets.

While I am not one to challenge the wisdom of those studios that sell their games by the millions, I nevertheless suggest that a character's movement can pull the player right out of the game experience if it is poor, and this is most certainly an area where future games are going to require more effort.

In terms of how best to generate movement, the debate between motion capture and hand animation continues to be a fierce one. Each side has its pros and cons, but all things being equal, the player really doesn't care how the animation was generated. Players want animation to be smooth, to look natural, and most of all to retain a solid frame rate (a separate issue entirely).

There is a threshold beyond which the detail included in a character's movement will go unnoticed by the player, or even become a hindrance. As with many noble causes, pursuit of truly realistic movement in games is both unnecessary and to some extent unwanted. Just as fully accurate real-world physics can grind a game into an unplayable pulp, providing a truly accurate movement system is unlikely to be responsive or fun. The trick is in achieving the right balance.



Characters

Character design is another core skill in a game artist's repertoire, and it's another area where getting it wrong is a whole lot easier than somehow landing upon the next speedy blue hedgehog or unfeasibly built female archaeologist. For all the effort that goes into game character design, there are some complaints that crop up quite regularly in this area.

Faces. As with the preponderance of poor facial animation in games discussed earlier, many artists complain that character's faces are often poorly realized. The type of game naturally has a lot to do with the emphasis put on characters' faces, but the bland, photo-mapped standard in many games can look out of place. Using a scan or digital photo as a starting point can make facial textures appear washed out. Additional work in Photoshop to increase the contrast and bring out the simple detail in the features can help a lot, even if the detail in the geometry isn't particularly high.

Stereotyping. It's not hard to think of some fairly pronounced stereotypes that run through the game industry's character designs. Naturally, a successful character leads to many imitations, but beyond that, the perception that our target audience is socially inadequate 14-year-old boys has led to forest elves dressed as Playboy Bunnies in leafy loin cloths and hardened female secret agents packing more silicon than a warehouse full of PCs.

There are reasons why character design in games follows this path, and while I could spend time suggesting that in the long-run it is neither healthy nor creatively satisfying to go down this route, I will say that stereotyping is both dull and unlikely to make our product distinguishable from the hordes of competition if we don't make more effort to innovate.

Variety. If a zombie is the reanimated corpse of a human, why do they all look the same? Did the world in which the game is set use vast amounts of cloning so that the living population all looked identical? Did some draconian law force them to wear the same clothing? Or is it just the same four meshes repeated 100 times?

I know once again that there are reasons why we don't get the chance to make a satisfactory number of different characters, but in certain types of games this can damage the player's experience and reduces the overall quality of the title. Visual variety is one of the best weapons an artist has to combat in-game monotony.

Next month, I'll look at some of the problems I've highlighted as they pertain to the areas of geometry, texturing, lighting, cutscenes, and effects.

GameDevelopers Conference Mobile

March 4 – 5, 2003 • San Jose, CA

MOBILE GAMES will change how consumers interact with technology, and the time to capitalize on this change is now. Estimates for mobile game revenue vary from optimistic to aggressive, yet major debate remains regarding the adoption of next generation technology, platform standardization, and viable business models. To profit from these changes, two worlds must come together.

Keynotes

Takeshi Natsuno
Executive Vice President
NTT DoCoMo i-mode

Ilkka Raiskinen
Senior Vice President
Media & Entertainment, Nokia

GDC MOBILE

www.gdcmobile.com

ONLY AT GDC MOBILE will leaders from the mobile communications industry meet with the world's best game developers. Network operators, content aggregators, technology and infrastructure providers and leading game developers will share information, explore opportunities and discover how to deliver compelling mobile entertainment to millions of consumers worldwide.

GDC Mobile sessions include:

- Beyond the Download: What's Next for Mobile Game Design
- International Case Studies from China, Finland, and the US
- Mobile Operator Roundup
- Betting on a Winner: What Makes a Great Mobile Game?
- Who Needs a Publisher? Who Needs an Operator?
- Multiplayer Games and Nomadic Communities
- Where are the GPS Games? Developing Games Using Geo-Location
- World Tour of Mobile Games and Interactive Entertainment

For more information on GDC Mobile speakers, sessions and events, go to

www.gdcmobile.com

Sponsors:

NOKIA MOTOROLA

GamaNetwork

MAKE BETTER GAMES

www.gdconf.com

Game Developers Conference 2001
Conference: March 5-9 • Expo: March 7-9
San Jose, California

Join the world's leading developers to set the standard for the next generation of games. Five days of intensive sessions provide information and inspiration in visual arts, game design, programming, audio, business and legal, and production.

Keynotes, tutorials and more than 300 lectures, panels and roundtables explore timely and relevant topics and processes in game development.

TUTORIALS

More than 25 tutorials including:

- **Digital Painting and Sketching** Donald Seegmiller
- **Making Characters Move: Expressive Character Acting through Laban Movement Analysis** Leslie Biskho, Jana Wilcoxen
- **Effective Project Management** Randy Angle & William Dwyer, Pronto Games
- **Built to Move in Max 5** Paul Steed
- **Direct 3D Advanced** Richard Huddy, Jason L. Mitchell, Guennadi Riguer & Alex Vlachos, ATI
- **Faster Math Functions** Robin Green, Sony Computer Entertainmet
- **All Stories Great and Small** Lee Sheldon
- **Running Massively Multiplayer Games** Raph Koster & Rich Vogel, Sony Online Entertainment, Gordon Walton, Maxis
- **Programming Games with Shockwave and Flash** Gary Rosenzweig, Clevermedia
- **Developing a Play Balance Plan** Prodipto Roy, Microsoft
- **Strategic Planning for Game Developers** Tobi Saulnier, Vicarious Visions
- **Creating Complex Character Rigs In Maya for Games** David Stripinis
- **Advanced OpenGL Game Programming** Cass Everitt, Simon Green & John Spitzer, Nvidia, Richard Huddy & Bill Licea-Kane, ATI, Evan Hart
- **Essential Mathematics for Games Programmers** Jim Van Verth, Red Storm Entertainment
- **Making and Managing the AAA Game Development Deal** Jim Charne, Dan O'Connell, Dan Offner, Ted Price, David S. Rosenbaum
- **Art and Science of Brainstorming** Noah Falstein, The Inspiracy
- **DirectMusic Tutorial** Scott Selfon, Microsoft
- **Game Tuning Workshop** Marc LeBlanc
- **IGDA Academic Summit** Presented by the IGDA
- **Microsoft DirectX Developer Day Sponsored Tutorial** Dave Aronson, Dave Bartolomeo, Chas Boyd
- **XBox Open Sponsored Tutorial**
- **>>GameOn! RenderWare and Criterion Software Sponsored Day**

PROGRAMMING

50 sessions including:

- **Hardware and Pipeline Optimization in Panzer Dragoon Orta** Takashi Atsu, Smilebit
- **Spherical Harmonic Lighting: The Gritty Details** Robin Green, Sony Computer Entertainment America
- **Dynamic Range** Paul Debevec, USC Institute for Creative Technologies
- **The Technology of Jak & Daxter** Stephen White, Naughty Dog
- **Optimized Stenciled Shadow Volumes** Mark Kilgard, Nvidia
- **Beyond Bouncing Boxes: The Physics of Things that Bend and Break** Jeff Lander, Darwin 3D
- **Memory Optimization** Christer Ericson, Sony Computer Entertainment America
- **Real-Time Fluid Dynamics for Games** Jos Stam, AliasIWavefront
- **Displacement Mapping** Michael Doggett, ATI Research & Tom Forsyth, Mucky Foot

GAME DESIGN

50 sessions including:

- **Dynamics for Designers** Will Wright, Maxis
- **FilmGame: Adapting Lord of the Rings** Neil Young, Electronic Arts
- **How to Make Your Game Successful in Japan** Tsurumi Roppyaku, 0600des gn & Ryoichi Hasegawa, Sony Computer Entertainment Inc.
- **Old School: Arcade Games in the Golden Age** Mark Cerny, Cerny Games, Eugene Jarvis, Ed Logg, Tim Skelly
- **Orthogonal Unit Design** Harvey Smith, Ion Storm
- **Designing a Ratchet & Clank Level** Brian Algier, Insomniac Studios
- **Achieving the Cinematic Feel for Medal of Honor: Allied Assault** Rick Giolitto, Electronic Arts
- **Designing Power-Ups for Action Games** Randy Smith, Ion Storm
- **Making Games More Fun: Methods for Playtesting Games** Bill Fulton & Kevin Keeker, Microsoft

Keynotes

Audio
Dane Davis
President
Danetracks

Production
Bruce McMillan
Executive Vice President,
Worldwide Studios
Electronic Arts

Business & Legal
Jake Song
Vice President and Chief Creative Officer
NCSoft

Game Design
Warren Spector
Studio Director and Executive Producer
Ion Storm

Programming, and Visual Arts Keynotes to be announced.

AUDIO

20 sessions including:

- **Audio Programming for Very Large Projects**
  Dave Ranyard, Sony Computer Entertainment Europe
- **Design With Music In Mind: Integrating an Immersive Game Score with Your Game Design**
  Guy Whitmore, Music Design Network
- **Geometric Modeling of Sound Propagation in 3D Games**
  Thomas Funkhouser, Princeton University
- **Physics-Based Synthesis of Sound Effects**
  Perry Cook, Princeton University

IGDA

25 sessions including:

- **AI Interface Standards: The Road Ahead**
  Alexander Nareyek, Carnegie Mellon University
- **Findings of IGDA's 2002/2003 Online Games Whitepaper**
  Alex Jarett, Broadband Entertainment
- **The Social Significance of Games: Killing Our Monsters**
  Gerard Jones
- **Videogame Violence: Rights and Responsibilities**
  Daniel Greenberg, Universal Interactive
- **What Developers Want: Industry Focused Curriculum Feedback**
  Jason Della Rocca, IGDA & James Thrush, Super X Studios

BUSINESS & LEGAL

25 sessions including:

- **Completion Bonding for Game Development Projects**
  Jason Kay, Society Capital Group & Frank Pape, Electronic Arts
- **Going Global: Effective Strategies for Protecting Intellectual Property Worldwide**
  Scott Pink & Peter Leal, Gray Cary Ware, Freidenrich
- **When the Site Hits the Fan: Wrangling Community Fansites**
  Elonka Dunin, Simutronics Corp. & Michael Perry, Maxis
- **How to Take Your Mobile Game to Market**
  Matthew Bellows, Wireless Gaming Review
- **Korea, Where Multiplayer Gaming is King**
  Robert Garriott, NCSoft
- **Mobile Games: Japan and the Future**
  David Collier, gamelet.com

VISUAL ARTS

35 sessions including:

- **Advanced Techniques in Creating Tile-Based Graphics for GameBoy Advance**
  Karthik Bala & Steve Derrick, Vicarious Visions
- **Character Rigging and Animation**
  Martin Coven & Martin L'Heureux, Inevitable Entertainment
- **Creating Aesthetic 3D Organic Volumes and Using the Right Tools for the Project**
  Lex Story, Microsoft
- **How to Build a Better Cutscene**
  Adam Schnitzer, LucasArts Entertainment
- **Stepping Away from the Pack: Resources and Methods for Developing Art Styles that are Memorable and Attention-Grabbing**
  David Rose, Red Storm Entertainment

PRODUCTION

30 sessions including:

- **10 Great Reasons You Don't Want to Make a Massively Multiplayer Game!**
  Gordon Walton, Maxis
- **Advanced Documents for Project Management**
  Dave Rohrl, EA/Pogo
- **How to Communicate with the Mothership in Japan**
  Ken Lobb, Microsoft & Jason Rubin, Naughty Dog
- **Managing a Large Scale Online Gaming Community**
  Raph Koster & Rich Vogel, Sony Online Entertainment
- **Managing the QA Process**
  Chuck McFadden, LucasArts Entertainment

REGISTRATION

- Get updated information on more than 300 sessions
- **Register by February 5, 2003** and **save up to 40%** on GDC Conference Passes

www.gdconf.com

GameDevelopers Conference

Voices of Reason

Advancements in game audio have brought us surround sound, epic film-style soundtracks and an increasing amount of character dialogue and narration. As a result, the need for professional voice talent is on the rise.

Who are "professional voice talent"? Professional voice talent come in two basic flavors, union and non-union. Union talent are members of the Screen Actors Guild (SAG) or the American Federation of Television and Radio Artists (AFTRA). In addition to providing members with health and retirement benefits, these unions provide a base pay scale and regulate working conditions. Anyone not belonging to the union would be considered non-union talent.

Does it matter if my voice talent is union or non-union? Creatively it makes no difference. There are thousands of great sounding voice actors to choose from. Union status comes into play with regard to pay rates and contracts.

Union members work from a standardized pay scale. This pay scale differs depending on the project type. For example, union rates for a commercial radio spot differ from those for an interactive media project. These rates and other union information are published on the SAG (www.sag.org) and AFTRA (www.aftra.com) web sites.

Videogames currently fall under the interactive media contract. This contract sets the rates and guidelines for using union talent on any interactive CD-ROM/DVD-ROM, Internet, or electronically published entertainment product. Union talent must be hired through a signatory. A signatory is a person who has registered with the union and has agreed to the pay scales and terms outlined in the contract. In-house voice recording using union talent requires the audio director or producer to contact the union office about obtaining signatory status. There is no charge for becoming a signatory. If you are contracting your voice-over casting or recording, check with your contractor because many casting agencies, talent agencies, and independent audio production houses are already signatories.

Choose the voices that bring the most depth and personality to your characters

Financially, union talent must be paid as employees of your company; taxes and benefits must be withheld. Smaller developers and publishers may find this process complicated. However, to ease this burden many producers employ the use of a paymaster. A paymaster is an independent payroll company or individual acting as the talents' employer of record, handling all of the talent payroll aspects for the project. Many paymasters also function as union signatories. Again, for those that are contracting their voice work, many contractors are already affiliated with a paymaster and it is frequently possible to pay the talent through them. It is worth noting that the interactive media contract does not require royalties or residuals. You will only be responsible for the talent's one-time session fee.

What about non-union talent? Non-union talent have their own individual pay rates and will negotiate their own contracts and payment terms. Their rates are usually flexible and project based. Some charge hourly, while others charge by the number of character voices performed. Be prepared to discuss residuals, call-back rates (should you need to recall talent for script changes), and game credits (both in-game and within documentation) with your talent. These issues might arise when negotiating the contract.

Where can I find voice talent? If you are contracting your voice recording work, chances are your contractor will offer casting services and can provide you with access to local and national union and non-union talent. Other options include talent agencies, theater companies, and television and radio stations. The Internet also hosts various free talent directories offering downloadable talent demos.

Two common misconceptions generally affect casting decisions: non-union talent possess inferior ability to their union counterparts, and union talent are always more expensive than non-union talent. Choose the voices that bring the most depth and personality to your characters, while maintaining your budget. Always cast creatively and intelligently.

MIKE VERRETTE | *Mike is the audio director for Wicked Noise, a recently founded game audio facility. When not listening to the voices found inside his head, he is usually recording the ones found outside. He can be reached at mike@wickednoise.com.*

SOUNDMAX®
SMartTools

GameDeveloper FRONTLINE FINALIST '02

CROSS PLATFORM

AUDIO MIDDLEWARE

CONTENT CREATION TOOLS

Create immersive sound effects by using advanced audio algorithms to deliver hyper-realistic game experiences. Sculpt, manipulate, and control complex audio behaviors and events in real-time.

Get SMart. Sound Better.

SoundMAX SPX Partners

FOX INTERACTIVE™

IMAGINE

INTRINSIC

motoGP.com
All Motorsports All The Time

TERMINAL REALITY GAMES

VALUSoft

VICTORY

Whispering Tree STUDIOS

DANGER
PIRANHA GAMES INC

DSI
Delphine Software International

SoundMAX SPX

www.audioforgames.com

ANALOG DEVICES

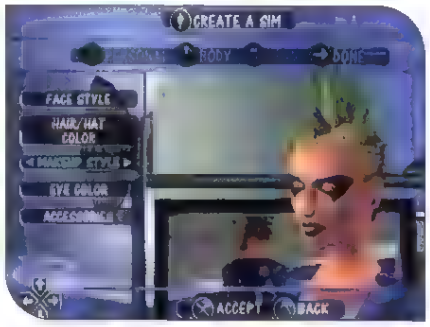
SoundMAX® is a registered trademark of Analog Devices, Inc
All other trademarks and registered trademarks are property of their respective owners. All rights reserved

AI Without Pain

This month I'm going to touch on four simple rules I've learned about game AI. Early in my career, I was a programmer and often had to make opponents seem intelligent and purposeful on very limited platforms. Making a virtue of necessity, I found there are several basic ways one can tweak a design to maximize the impact of AI while minimizing the difficulty and complexity of the code. In keeping with this column's overall goal of simplicity, these rules will abstract the usual format of this column and stick to the fundamentals.

AI Rule #1: Make the effects of the AI visible to the player. It can be tempting to model subtle choices in your AI, but unless the final results are clear to the player, you may well be wasting your time. You can choose to model clearly visible choices; for example, a possible Sims mate can touch your character's arm and laugh, or turn a cold shoulder. Or to flip that around, you can alert the player directly when a subtle choice is made; for instance, when an enemy sniper is responding to a player's choice to run straight ahead instead of crawling stealthily around the flank, an audio cue like, "Look, there he is!" lets players know the AI is on to them.

AI Rule #2: When simulating a real system, use real-world formulas and cheat as little as is feasible. This is a tough rule to follow, and "feasible" can be very subjective. Still, avoiding early shortcuts often saves time in the long run. It can be tempting to cut corners with canned animation or table-driven behaviors, but if you begin with real formulas to simulate real-world consequences, you'll find that later expansions to the AI that also stick with actual physics can fit in seamlessly. For instance, in a racing game it may be tempting to have an AI car jump a gap with a preprogrammed animation, but if an opponent's race car is subject to the same constraints as a player's car when jumping a gap, the level designers can add new jumps or adjust old ones without having to go back and change all the previous enemy behaviors.



Character qualities in THE SIMS suggest subtle motivations in players' imaginations, following AI Rule #4

This rule can be trumped when a simplifying shortcut can save huge development costs, or be trumped by the previous rule when the difference between a shortcut and actual calculation is invisible to the player.

AI Rule #3: Add a small amount of randomness to your AI calculations. A little randomness can make a dumb AI look very smart. Enemies that respond exactly the same every time feel robotic and predictable. But just 5 percent variation can shock a player out of complacency and make an opponent seem alive. Sometimes the easiest way to achieve apparent randomness is to add plus or minus a few percent to a basic calculation of distance or direction. This technique is particularly effective for animal behavior.

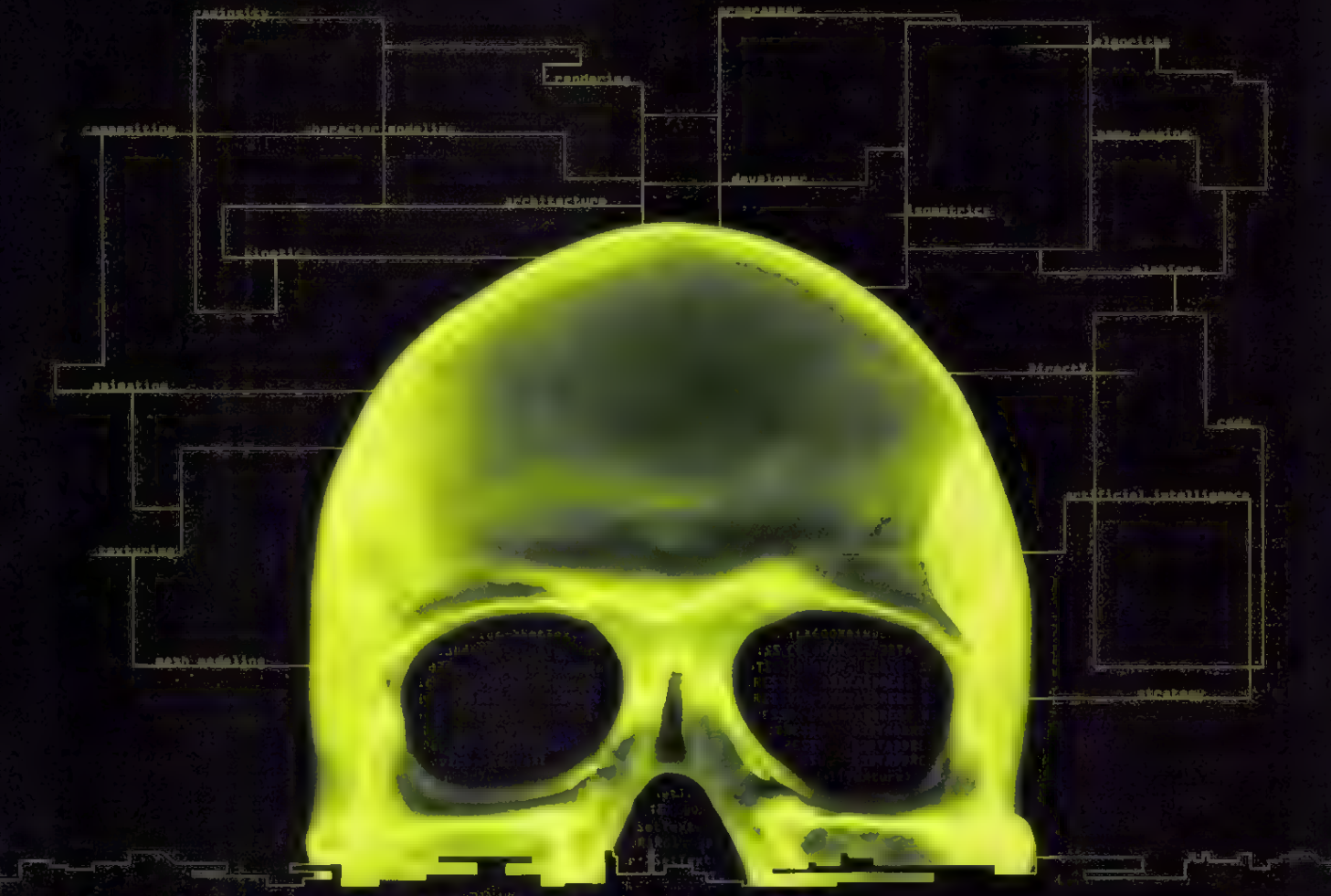
How much randomness is enough? It should be at least enough to satisfy AI Rule #1; if the random numbers don't make any measurable difference to the player, then there's no point to implementing randomness. And there should not be so much randomness that the AI seems wildly unpredictable or unfair. Often a little goes a long way.

AI Rule #4: Create the AI in the mind of the player through suggestion. The ideal AI implementation is not actual intelligent behavior but rather the illusion of intelligent behavior. Much like Sun Tzu's precept in *The Art of War* that the best way to win a battle is to make fighting unnecessary, the best way to provide AI is to let the players imagine it without a need for coding. Simply implying that special behavior might occur can plant it in your player's imagination. Call an enemy unit "elite" and give it a special color and players will treat it differently, crediting it with superior abilities. The audio track in the original MEDAL OF HONOR made it seem as if squads of German soldiers patrolled just beyond the limits of the player's vision. I'm sure you've seen other examples.

When designers combine this rule with the previous rule, players can project complex motivation onto simple, random behavior. In one flight simulator I worked on, we mentioned an enemy formation flying feature that, through a last-minute change, didn't make it into the game. We still got letters from people about it, uniformly praising us for the effect. It was an unintentional omission, but ultimately a very cost-effective way to add the illusion of intelligence.

Sneaky? Perhaps. But if the end result is more enjoyment for less effort, I'm willing to risk it.

NOAH FALSTEIN | *Noah is a 22-year veteran of the game industry. You can find a list of his credits and other information at www.theinspiracy.com. If you're an experienced game designer interested in contributing to The 400 Project, please e-mail Noah at noah@theinspiracy.com (include your game design background) for more information about how to submit rules.*



Game Development. It's Serious Business.

"Game programming is without a doubt the most intellectually challenging field of Computer Science in the world. It's the fusion of art, science, mathematics, fiction, and more. A game programming book should be an exciting adventure for both the author and the reader."

—André LaMothe, Series Editor

The Premier Press Game Development Series.

- Game Design: The Art & Business of Creating Games
- 2D Artwork and 3D Modeling for Game Artists
- Beginning Direct3D® Game Programming
- Data Structures for Game Programmers
- Game Scripting Mastery
- OpenGL Game Programming
- Swords & Circuitry: A Designer's Guide to Computer Role-Playing Games
- The Zen of Direct3D® Game Programming
- Multiplayer Game Programming

Premier Press

Premier Press
A Division of Course Technology
www.premierpressbooks.com

Available at your favorite bookstore or call 1.800.842.3636 to order.

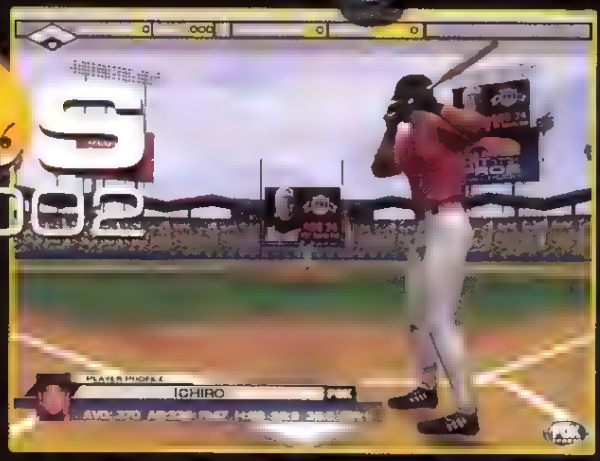
SSX SNOWDREAMS

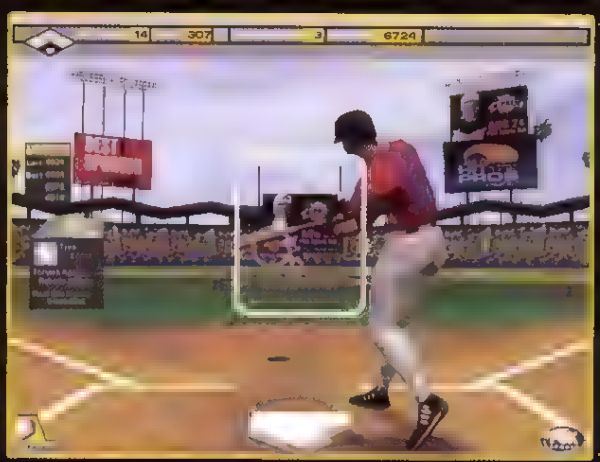
EA.com
ELECTRONIC ARTS

HIT THE PROS 2002

FOX SPORTS

www.hitthepros.com





Developed By: sandlot games

Developed By: FAUST LOGIC INC

Want to develop great online games AND earn meaningful revenues? Sandlot Games and Faust Logic did and now their Web Driver-powered games are being distributed to an audience of over 30 million via WildTangent's distribution network of 20+ portals, ISPs, and OEMs. To learn more about the power of WildTangent's Web Driver platform and distribution network, contact us at: techlicense@wildtangent.com

WildTangent.

PIONEERING INTERACTIVE BROADCAST GAMES

www.wildtangent.com

The 2002 Front Line awards

Game Developer's 2002 Front Line Awards mark the fifth year that we've set out to recognize the very best in the world of game development tools and are perhaps the most diverse group we've yet assembled. The winners range in price from many thousands of dollars all the way down to free, and they target game projects from high-end PCs to mobile phones.

The selection process for the awards began with open nominations from the game development community last August. Anyone could nominate any game development related product released between September 1, 2001, and August 31, 2002, via the *Game Developer* web site. Response was plentiful and passionate. Our panel of judges then winnowed this worthy but very large field of nominees down to a group of finalists for each award category. After hands-on evaluations, winners were determined by a balloting process that weighed such product considerations as utility and integration, ease of use, interface, cost, and innovation.

A few changes were made to our presentation format for 2002, as the Production category made way for a new award for Game Components. The inclusion of this category reflects the growing importance of middleware in the industry. Nominees in this category included game components for networking, rendering, and physics. Despite the disparate nature of the finalists, it proved to be among the most competitive of our categories.

As always, the annual Hall of Fame award recognizes a game development tool that has been in release for at least five years and that has made an indelible impact on how games are made. This year, Microsoft's little set of libraries called DirectX joins past inductees including the 3dfx Voodoo card, PC-Lint, Pro Tools, Watcom C/C++, Borland C/C++, Miles Sound System, BoundsChecker, SoftICE, 3D Studio 1.0, Deluxe Paint, DeBabelizer, Sound Forge 1.0, Cool Edit 1.0, Cakewalk Pro for Windows 1.0, Sound Blaster 1.0, and the books *The Mythical Man Month* and *Computer Graphics: Principles & Practice*.

These awards wouldn't have happened without our Front Line Award judges, all of whom graciously gave us time they didn't have (judging took place at the height of the holiday crunch) to thoughtfully examine and critique every game development tool they received. The Front Line Awards aim to honor innovation and quality in game development tools, and the judges took that goal to heart as they recognized achievements made by both new technologies and venerable tools that have consistently raised the bar. So to our pool of judges, whom we selected for their talent and admire for their dedication to this industry, we offer sincere thanks.

Programming: Ralph Barbagallo of Flarb Development, Chris Corry of LucasArts, Mark DeLoura of Sony Computer Entertainment America (and former editor-in-chief of *Game Developer*), 3D programming consultant Ron Fosner, Herb Marselas of Microsoft, and Matt Pritchard of Ensemble Studios.

Art: Tom Carroll of Vision Scape Interactive, Spencer Lindsay of Etribe Studio, Kian Bee Ng of Polyphony Digital, and Todd Siechen of Real Eyz Imaging.

Audio: Chuck Carr of Sony Computer Entertainment America, Tom Hays of Treyarch, Aaron Marks of On Your Mark Music Productions, Gene Porfido of Smilin' Pig Productions, and Rob Ross of Sound Endeavours.

Game Components: Bill Allen of Datadesign, Karthik Bala of Vicarious Visions, Jonathan Chey of Irrational Games Australia, Charlotte Chiang of Game Factory Digital Entertainment Studios, Gary Foreman of Rockstar Games, Dennis Harper of Pacific Coast Power and Light, Robert Huebner of Nihilistic Software, and Mike McCool of Slightly Subtle Technology.

The Hardware and Hall of Fame categories were both balloted by judges across all of the categories.

—Daniel Huebner

2
1
3

hall of fame

Microsoft
www.microsoft.com/directx

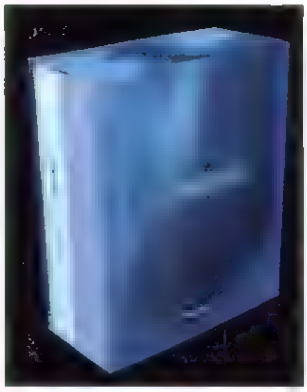
DIRECTX

Love it or hate it, no one can deny that DirectX is proving to be a critical force in forging a viable market for high-end PC gaming. DirectX's 1995 first release drew mixed reaction; developers found the API to be a bit bulky and arcane.

The growth of 3D graphics hardware for PCs was the deciding moment for DirectX adoption. Each hardware vendor supported their own proprietary 3D API, and this fragmented the market for several years. But with each new revision, DirectX — and Direct3D in particular — kept pace with the hardware vendors, eventually convincing all but the most die-hard developers and hardware vendors to support their standards. With the release of Dreamcast and later Xbox, DirectX's reach spilled beyond the PC world and into console gaming.

DirectX's induction into the Front Line Hall of Fame came as much from persistence and politics as it did from technical wizardry. Microsoft's willingness to invest huge sums of money and time into a project which, quite literally, no one else could have attempted, has paid off handsomely.

— Robert Huebner



Discreet
www.discreet.com

3DS MAX 5

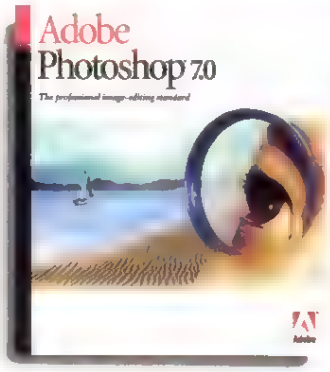
3DS max has come a long way since its inception and continues to be the standard for artists developing content for games. Discreet has done a wonderful job of listening to user demands and implementing features that are useful and efficient. Character animation especially has been improved tremendously with the release of 5.0 — I'm not sure if Character Studio is still even necessary, and I'm beginning to suspect that after a couple more version upgrades there won't be much need for after-market plug-ins at all anymore. Discreet continues to drive forward steadily, and the numerous new features in Max effectively address nearly all of the weaknesses of previous versions, which is why it deserves the Front Line Award.

— *Todd Siechen*

PHOTOSHOP 7

Photoshop comes as close as any tool to being standard equipment for game development. Out of the box, Photoshop 7 looks very similar to Adobe's last update, but closer inspection reveals new features that definitely improve the way game makers interface with this tool. While the new Brush Dynamics and the Healing Brush tools alone make the upgrade worth looking at (see our full review on page 11 of this issue), it's Adobe's commitment to continual improvement that earned Photoshop 7 a Front Line Award.

— *Spencer Lindsay*



Adobe
www.adobe.com

BioGraphic Technologies
www.biographic.com

AI IMPLANT 1.0

Game animators are constantly faced with two challenges: One is the physical constraint — can the existing hardware allow you to create your animation at an interactive rate? The other is the algorithmic constraint — can your algorithmic model accurately depict the intended result? Making large, controlled scenes is a prime example of these challenges. The beauty of AI Implant's AI driven character animation toolset is that it provides artists a means to animate large scenes by splitting them into manageable parts as necessary, and yet the artist retains control over the individual or group characters as an integrated whole.

— *Kian Bee Ng*

3DS Max 5 — Discreet; AI Implant 1.0 — BioGraphic Technologies; DarkTree 2 — Darkling Simulations; Deep Paint 2.0 — Right Hemisphere; Lightwave 7.5 — Newtek; Messiah: Animate 3 — Project Messiah; Photoshop 7 — Adobe; Realviz Interactive Studio — Realviz; SoftimageXSI 2.0 — Softimage

game components

Havok
www.havok.com

HAVOK

When it comes to middleware, there are easy problems and there are hard problems. Physics is a very hard problem. No two game projects are exactly alike; different platforms, different tools, and different coding standards abound, yet great middleware has to achieve drop-in integration with any project to be a success. With something like a video playback library or a sound mixer, this is a reasonable goal. But with something like a physics engine, it borders on impossible.

Havok has achieved success in this difficult arena by employing excellent engineers and support staff, both in Europe and the U.S., and by constantly improving their product in response to developer feedback. The first few versions were rough — it was a memory-hungry system that was focused largely around PC racing game physics. But in the last couple of years, Havok has pushed hard in areas that are important to gaining broader appeal — aggressive memory savings, more attention to character-based games, and focusing on the core physics that developers want most. If the trend continues, maybe we'll be able to move physics into the "easy" column at last.

— Robert Huebner

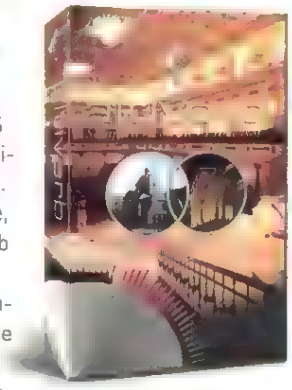
Havok 1.6 — Havok; Intrinsic Alchemy 2.0 — Intrinsic Graphics; Netimmerse 4.2 — NDL; Renderware Platform 3.3 — Criterion; Zona — Zona Inc.

audio

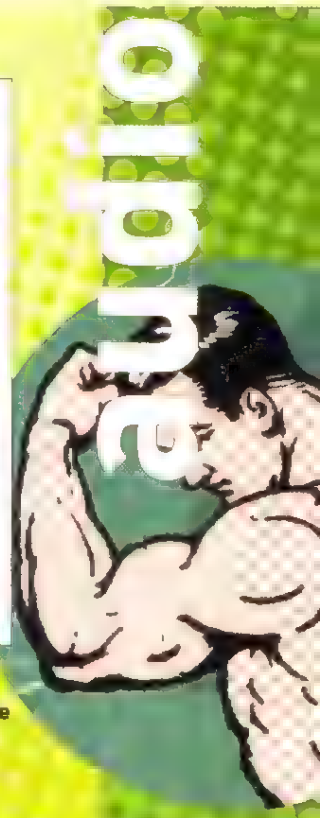
ALTIVERB 2.0

Altiverb 2.0 from AudioEase is a "sampled-acoustics" reverb plug-in for the Macintosh that supports every major plug-in format, including MAS (Digital Performer), HTDM, RTAS (Pro Tools), and VST. Altiverb offers the ability to add real-world reverb spaces, rooms, and environments to audio tracks. Using impulse responses generated by a starter pistol or sweeped sine wave, Altiverb allows the user to sample virtually any space and turn it into a reverb by a process called convolution. This method replaces the sampled spikes with the impulse responses, creating a reverb model incredibly real and natural sounding, much more so than your usual synthetic reverb processors. One to four channels, including surround capabilities of incredibly realistic sampled rooms — from world-renowned halls and cathedrals to closets and bathrooms — can add depth to music and sound effects that can only be matched by hardware devices costing up to $10,000. Altiverb does it for under $800.

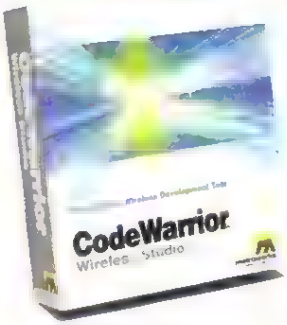
— Gene Porfido



Altiverb
www.altiverb.com



Altiverb 2.0 — Audio Ease; Garritan Strings — Harps.com Corp.; Kontakt — Native Instruments; Reason 2 — Propellerhead; Sonar XL 2 — Cakewalk; SoundMax SmartTools 2.0 — Analog Devices



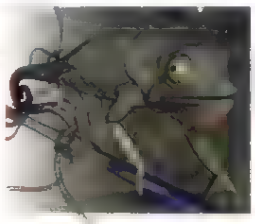
Metrowerks
www.metrowerks.com

CODEWARRIOR WIRELESS STUDIO

Metrowerks' CodeWarrior Wireless Studio is a mighty morphin' J2ME IDE which allows for the easy compilation of Java code for a wide variety of different Wireless Java implementations. Of course, the robustness of the SDK integration is directly related to the quality of the third-party vendor's own J2ME kit. Regardless, CodeWarrior does an impressive job of easing the nightmare of multiple handset integration while still packing industrial-strength J2ME IDE features at a relatively low cost. The added bonus of an on-handset debugger will become invaluable once more handset manufacturers support the feature in their development firmware.

— *Ralph Barbagallo*

REAL-TIME RENDERING, 2ND EDITION



I can't think of any higher praise for a book than the fact that it's always on my desk and within easy reach, and *Real-Time Rendering 2nd Edition* is one of the few books that qualifies for that distinction. *Real-Time Rendering* provides thorough coverage of the current state of the art in real-time graphics, as well as case studies, appendices to help brush up your math skills, and a voluminous source bibliography. There's no doubt that this is a must-have volume for any graphics programmer.

— *Herb Marselas*

1 Multiple Bump Layers
2 2D Glare Halo

Nvidia
www.nvidia.com

Cg

For some time now the programming of embedded graphics co-processors has been a black art, closed off to all but the most dedicated practitioners of the 3D coding craft. Cg changes this landscape with one fell swoop, allowing programmers to use a relatively high-level C-style language to build hardware-accelerated vertex and pixel shader programs. As instrumental as Cg promises to be in exposing a larger audience to low-level graphics programming, what makes Cg even more exciting is its potential to dramatically reduce the learning curve for designing and building very complicated shader effects. Coupled with Nvidia's unusually inclusive efforts to keep Cg API- and hardware-agnostic, the language has a more than fighting chance of emerging as a cross-platform de facto standard (although all you PS2 vector unit programmers probably shouldn't hold your breath). While Cg still has some high expectations to live up to, its audacious promise and innovative spirit is more than enough to earn it a coveted Front Line Award.

— *Chris Corry*

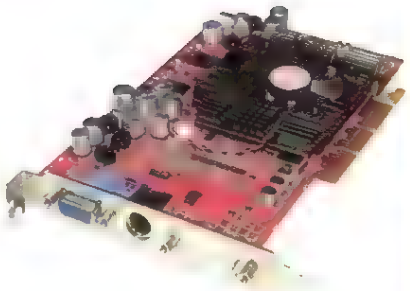
Cg programming language — Nvidia; CodeWarrior Wireless Studio — Metrowerks; *Game Programming Gems 3* — Charles River Media; Test Track Pro — Seapine Software; Great Circle 6.0.0.9 — Geodesic; InstallShield Multi-Platform 4.5 — Install Shield Corp.; PS2 VectorC {EE} PS2 Compiler — Codeplay; *Real-Time Rendering*, Second Edition — A.K. Peters; SN Systems Tuner for PS2 — SN Systems; Vtune 6.1 — Intel

hardware

ATI RADEON 9700

The ATI Radeon 9700 receives a Front Line Award not only for being a superb piece of hardware, but also in recognition of ATI's leadership in promoting advanced graphics education and capabilities. ATI has emerged as a leader in bringing not only advanced, robust hardware such as the Radeon 9700 to the market, but also by promoting education and tools for developers and artists to help them understand and take advantage of the features. These efforts are well spent, considering the 9700 brought the first DirectX 9–compliant graphics card to market prior to the official release of DirectX 9, allowing developers to evaluate DirectX 9 features for themselves.

— Ron Fosner



ATI
www.ati.com

WACOM CINTIQ 18SX

I've played with LCD pen tablets before, but Wacom's Cintiq 18sx changes the way I interface with my computer. Combing an 18-inch, high-resolution LCD screen with 512 levels of pressure sensitivity and very fast response times, the Cintiq has the effect of removing even further the interface barrier between me and my artwork. Using this thing with the new brush modes in Photoshop 7 is amazing, like painting with a real brush.

— Spencer Lindsay

Wacom
www.cintiq.com

3DBoxx — BOXX Technologies; Cintiq 18SX — Wacom; GeForce 4 Ti — Nvidia; NT 4 Microphone — Rode; Precision M50 Mobile Workstation — Dell Computer; Radeon 9700 — ATI; Wildcat VP — 3Dlabs

Rewarding Innovation in Independent Games

See the 5th Annual Independent Games Festival Competition finalists at

www.igf.com

Quick Facts About the 2003 IGF Competition

Entries:	73
Average deve opment cost:	$36,770
Average development time:	14 months
Number of games with $50,000+ budgets:	10
Number of games with $0 budgets:	25
Number of games in development for ess than 1 month:	3
Number of games in development for more than 2 years:	9
Percentage of games developed by teams with high hopes for fame and fortune:	100%

The 5th Annual Independent Games Festival
March 6-8, 2003 - San Jose, California

Craft sponsor, Innovation in Game Design
Microsoft DIRECTX

Craft sponsor, Technical Excellence
intel

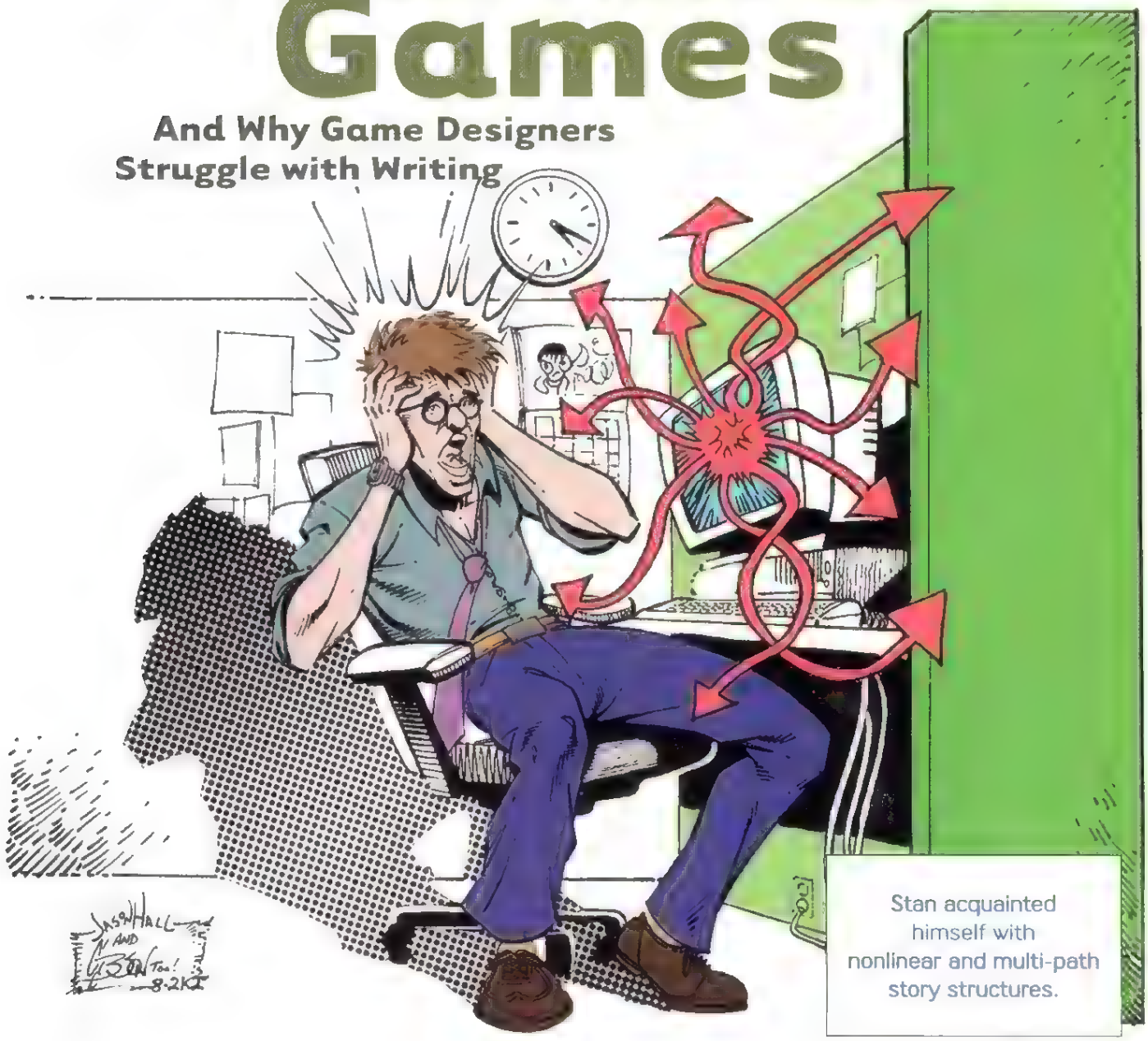
Josh and Dan Goldstein, members of Shizmoo, accepting the Audience Award for KUNG FU CHESS at the 4th Annual IGF.

INDEPENDENT GAMES FESTIVAL

GamaNetwork

What Screenwriters Don't Know About Games

And Why Game Designers Struggle with Writing



Stan acquainted himself with nonlinear and multi-path story structures.

DAVID FREEMAN | *David is currently working as a designer/writer on games for Activision and Midway. He also contributed to the script for Shiny's* ENTER THE MATRIX *game and helped Microsoft on a game concept. His book,* Creating Emotion in Video Games, *is due out soon by New Riders Publishing, and he is speaking on this subject at the 2003 Game Developers Conference. David also teaches the screenwriting class "Beyond Structure" (www.beyondstructure.com), which has attracted many game designers and game producers. As a writer and producer, David has had scripts and ideas bought or optioned by MGM, Paramount, Columbia Pictures, Castle Rock, and many other film and television companies. Contact him at freeman@dfreeman.com.*

Having kicked around inside the worlds of both films and games, I've come to appreciate the gap between them. It has given me an interest in isolating certain aspects of advanced character creation and storytelling that can, with considerable reworking, be applied to games.

This process alerted me to a current impasse in games: On one hand, many game developers want to improve the dimensionality of their characters, the depth of their stories, the emotional spectrum of the game experience, and the overall writing style in their games. On the other hand, bringing in a professional writer from outside can be problematic. Even with extensive screen credits and impressive writing samples, a writer may not understand what makes games unique as a medium, and thus may provide material of dubious value.

As games' production values continue to rise in order to meet consumer demand, many of today's projects are being allocated larger writing budgets, tempting some producers to hire big-name screenwriters. And while there's a lot of great talent to draw upon in Hollywood (if you have the budget), even experienced screenwriters will need help coming up to speed quickly on writing for games instead of for the screen. Identifying these differences early on will help your investment go a lot farther toward producing quality entertainment.

A Storytelling Medium Like No Other

Because film and TV are linear media, the screenwriter might believe that for a game, the player should follow a set route to make sure that the player experiences the whole story. You'll need to educate the writer to the fact that, normally, if there aren't options for players to escape from merely following the story if and when they wish, players will feel too much like the game is playing them, rather than they are playing the game.

The screenwriter is unlikely to realize that, even in games with stories, ways exist to play the game that completely avoid the story altogether. GRAND THEFT AUTO 3's "vigilante mode," in which players chase down criminals in a stolen police car for fun independent of the game's plot advancement, is a great example of this.

Therefore, the screenwriter needs to know dozens of other ways to make the game emotionally immersive so that it will be compelling even if the player never experiences the story or puts the story on hold for a while. For instance, how emotionally complex is the player's relationship to the NPCs? How much emotional depth can be crammed into a single line of NPC dialogue? How rich is the world of the game? Does the player, through his or her character, ever wind up in emotionally complex situations?

Screenwriters create emotional experiences in movies by making a story unfold in a particular sequence, whereas many games are designed so that a player might come upon different elements of the story in a variety of orders. Thus the writer needs to know how to create emotion in games that contain not only linear components, but also nonlinear (where a game can be played in a variety of different orders) and multi-path (where there are specific branching points of a story line) aspects as well. As many games intertwine all three modes, sometimes even adding other ways of experiencing "story" by giving the player a choice of roles to play, these different structures need to be understood, embraced, and artfully used as creative tools by the writer.

Some games are designed such that players might select their route through a mission or even choose which missions to run. In such situations, the player might even miss some pieces of the story altogether. The screenwriter needs to know how to retain the game's emotional impact even if the player completely skips whole sections of the story.

Creating Playable Roles

For my first few years after college, I spent my weekends in spring as a costumed singer at the gigantic Renaissance Faire held annually outside of L.A. I was one of about 2,000 employees and volunteers, all of whom willingly put their hearts and souls into speaking in dialect and acting out different roles from the English Renaissance period.

In a game, a screenwriter will often get excited about the idea of a player acting out a certain role in the story. But if the game, or the game's story, casts the player in a role — let's say a space pilot — that doesn't automatically mean that the player feels like a space pilot. You'll need to coach the writer on how to make the players willingly adopt their character's role.

The writer might think that casting the character as a hero solves this problem. Well, you may make them the hero, but that doesn't mean players won't go where they please, performing unheroic acts, blasting everyone and everything in sight, including orphans, babies, sweet old ladies, and hungry puppies who need a home. The screenwriter needs to allow the player this freedom but still give the player incentives to follow the story (if there is one).

One of the common terms kicked around screenwriting circles is "character arc." The character arc is a character's difficult path of emotional growth though the course of a story. At the start of *Star Wars: Episode IV*, Luke doesn't know who he is. By the end, he has become a Jedi knight, and he knows it. The path he takes getting there is his character arc.

Let's go back to our space pilot example. The screenwriter might want the player character to have an arc. For instance, at the beginning of the game the pilot is supposed to be a coward, but by the end of the game the pilot is supposed to be brave.

There are numerous problems here, and they go back to the fact that the player himself is part of the action. Just because the screenwriter says your character is a coward doesn't mean that you feel like a coward when you begin playing the game. And if all the other characters in the game treat you as if you're a coward, but you don't feel like one, this can sever your emotional bond to the game rather than enhance it. Similarly, just because the screenwriter thinks your character should feel brave at the end of the game doesn't mean you feel any braver than you did at the beginning.

It's not that what I call a "first-person character arc" is impossible to create. But you may need to point out to a screenwriter that the methods used to create character arcs in traditional scripts may not have much value in games.

Dialogue in Films vs. Games

In your game, you may want to have the player's character develop friendships and other relationships with NPCs. You'll need to explain to a screenwriter that most games don't allow for a lot of room for talk, as it can detract from the fun and momentum of gameplay. Thus, the screenwriter has to know how to create emotional bonding between the player and NPCs without a lot of speech.

Because dialogue is often minimal in a game, the writer needs to be a proven master at creating complex characters — characters that are not only complex but likeable as well — even if that character speaks few words. By proven, I mean exactly that: screenwriters should be able to prove that they can do this before you hire them.

Ah, So We Should Just Hire Comic Book Writers

I love reading certain comic books, and I understand why some game companies might turn toward comic book writers. After all, they have a demonstrable ability to convey story and characters with minimal dialogue.

However, my experience suggests that if the game requires complex and rich characters and stories, the skills and talents needed could very well exceed that of most comic book writers. (There are exceptions to this statement, and if you find one who's perfect for you, hire him or her.)

For instance, imagine trying to bring out the richness of the world, story, or characters of *The Lord of the Rings* in a comic book. For that matter, try to imagine bringing out the emotional complexity and nuances of the characters in *Buffy the Vampire Slayer* or *Angel* in a comic book.

A Different Kind of Process

In films, the writer comes up with an idea and then writes a script. In game design, this is rarely the way things happen. You'll need to find a writer who is comfortable creating as part of a group.

You should inform the writer that he or she needs to be extremely flexible. The screenwriter might specify that certain key pieces of information or aspects of a character will be revealed at a particular part of the game, only to learn later that the entire level where that piece of information was to be revealed has now been eliminated from the game altogether.

Similarly, separate levels are sometimes combined, or new ones are added as the game is being built. The screenwriter must adapt to this creative state of affairs easily and happily.

You'll also need to make sure they can work on a flexible schedule. They might be needed at some points in the game, and then, after the story and character descriptions are worked out, they may be sent away for three to seven months while the game continues to be built.

One of the hardest tasks ahead of you is training the writer to think more like a programmer. The writer needs to think not in terms of linear story but in terms of options. These include options as to what actions might be available to the player at any point in the game, or options as to what an NPC might say or do, and deciding which factors trigger when an option or set of options becomes activated. This way of thinking, so basic to a programmer, will likely be new to the writer. And yet the writer needs to do more than understand this way of thinking; these options are fundamental in creating the means by which game stories unfold.

Many game developers find their first encounter with Hollywood quite memorable

If You Like Fantasy Worlds, You'll Love Hollywood

In life, it ain't over until the fat lady sings. In games, it ain't over until all the NPC dialogue is written.

I was speaking to a high-placed executive at a large game company. He had just begun a relationship with one of the biggest and most prestigious Hollywood agencies and was very excited. I could almost hear his heart pound as he told me some of the famous writers to whom their agents had introduced him.

I asked him how much he'd have to pay these famous writers, and his voice lowered a bit as he said he was supposed to shell out hundreds of thousands of dollars, plus offer them all sorts of back-end revenue, in exchange for their story ideas.

I paused for a moment of sad contemplation, the way I always do when I imagine the innocent getting led to slaughter. Then I took a deep breath and explained how he was about to get fleeced. I pointed out the following.

First, the big-name writers he was talking about are always booked for several years out. This means that they'd take out a few weeks (at best) to whip together a story for him, take the hundreds of thousands of dollars, and then run back to their much more prestigious and lucrative screenwriting deals.

Second, because of the way games are developed and the degree to which they can change from concept to completion, who even knows how much of their story ideas would actually be present in the final game? And yet the game company would be out tons of money.

Third, because the most in-demand writers have the busiest schedules, the agency would probably end up trying to get the company to accept writers "from the bottom of the deck" — namely, writers who haven't worked for quite a while, often because they're not particularly good. And the game company would still be overcharged.

And finally, I asked whether the famous writers he was speaking about were going to stick around and write 1,000 or 10,000 lines of NPC dialogue, one line at a time. Or were they going to keep the exorbitant money they've been paid, grin as they walked away from their NPC-writing duties, and go back to making millions writing screenplays and soaking in the glamour of Hollywood? If he's hiring a famous writer, I asked the executive, exactly what is he getting for his money?

There was a long pause on the other end of the line, as the executive realized he was on the verge of becoming 21st century roadkill, blindsided by a 12-ton semi in a B-grade Hollywood shark-o-rama.

The game executive was no dummy. He had just never been a bit player in a real-life version of *Jaws* before.

If game publishers and developers are serious about spending big-time money on big-name writers, they need to examine in painstaking detail what they're getting for their money, or they may well end up feeling burned and used.

Where Writers Fear to Tread

It's unlikely that the screenwriter you hire will know that putting a player into a story, even an interesting one, is only one of many ways to engage the player emotionally in the game. After all, sports games and racing games have little story but can be very emotionally engaging.

Thus, there are many ways of creating emotional immersion in a game, ways that might not be in your writer's toolbox. For these, you might be on your own. They include:

- Integrating the story with the gameplay mechanics
- Making the player care about the world of the game
- Creating emotionally complex relationships between the player and the NPCs, and between the NPCs and other NPCs
- Figuring out how much and in what way the world of the game affects the player, and to what degree, if any, the player affects the world of the game
- Putting elements into the game that will motivate the player to continue through to the end, instead of stopping after a few levels.

Why Game Designers Struggle with Writing

In the screenwriting classes I teach, I've found that many new writers believe that, because they've watched innumerable films and TV shows, they can automatically write. I've frequently found the same assumption being made by some in the game business. At first, it seems like a natural conclusion.

However, that's like saying if you've listened to lots of rock music, you should be a guitar master. The reason almost no one becomes a great writer just by watching great films and TV is because writers use hundreds of different techniques to make their work emotionally engaging, almost all of which operate outside the audience's awareness. The more skilled writers are at concealing their techniques, the more engaging the film or show becomes.

Understanding how such techniques work is helpful whether you intend to develop your own talents as a writer or simply need to understand better where your hired screenwriter is coming from.

2:37 AM. ENORMOUS GAMES.
THE GAME SHIPS IN SIX WEEKS - BUT WE FORGOT THE DIALOGUE!
BUT A WRITER'S NOT IN THE BUDGET!
LET'S HIRE A COMIC BOOK WRITER!
THAT'S A GOOD IDEA. IN FACT...
IT'S MY IDEA!!!
THREE WEEKS LATER.
IT LACKS NUANCE!
METAL GEAR LIQUID 2
grand theft moped
SCRIPT

Crouching Tiger, Hidden Technique

Let me give an example of such hidden writing techniques. Many people enjoyed the movie *Crouching Tiger, Hidden Dragon* and were moved by its emotional depth and spiritual resonance.

But emotional depth of the plot was no accident. It resulted in part from the artful use of what I call "plot-deepening techniques." While the total number of plot-deepening techniques is extensive, *Crouching Tiger* beautifully utilized six of them:

1. Two major characters trade places. Chow Yun-Fat's character has left a monastery and his spiritual discipline to pursue both love and the role of an ethical warrior. Zhang Ziyi's character ultimately gives up her narcissism, her lover, and even her life for an act of spiritual redemption.

2. A spiritual power is made palpable. This is accomplished by showing Chow Yun-Fat's mystical and spiritual abilities, which exceed Zhang Ziyi's. His abilities are demonstrated in such moments as his repelling her fierce swordsmanship with a simple stick, in Zhang's inability to dislodge him from his serene pose on the tip of a long tree limb, and even in his continuing to care for Zhang despite her narcissism, for he sees the potential beauty of her soul — the potential of which she's totally oblivious.

3. A symbol takes on more and more emotional associations. At fist, the sword Zhang wants is simply a superior, maybe even magical weapon. But it comes ultimately to stand for full spiritual attainment. When Chow Yun-Fat takes the sword and throws it in the river, saying she has no use for it, he means she's not ready for spiritual truth.

4. A character we like (Chow Yun-Fat) dies.

5. A bittersweet ending is created when a character accomplishes their character arc but not their goal. Actually, this happens with three characters. Chow Yun-Fat's goal isn't to die, but before he does, he confesses his love and passion for Michelle Yeoh. Her goal isn't to lose him, but before she does, she also communicates her love. Zhang's goal in the film isn't to jump off a mountain, but before she does, she has her spiritual and ethical transformation.

6. The ending is a little open, leaving the audience to complete the story. We're told that when you jump off that mountain bridge, anything you wish for will come true. What did Zhang Ziyi wish for when she dove into the abyss? Each of us completes the story in our own way.

When watching the film, did you spot that some of the emotional impact of the

ABOUT THE ILLUSTRATIONS

Pages 42 and 46: pencils by Jason Hall (http://vanian.home.mindspring.com, vanian@mindspring.com), inking by Chuck Gibson (gibson_inks@hotmail.com)

Page 44–45: pencils and inking by Don Anderson (www.stormlab.com/Don, DLA5740@aol.com)

film was due to the artful use of these six plot-deepening techniques? Did you also catch the eight character- and scene-deepening techniques that added to the film's emotional power?

Don't feel bad if you didn't. Had you been able to pull back and become that analytical, the movie would have been so emotionally unengaging that it would have been a failure. The secret of great writing is utilizing the techniques that evoke emotion so artfully that they operate outside the audience's or player's awareness.

If you had hundreds of such techniques at your disposal, many wouldn't work in games, but about half would. Similarly, there would be many other techniques that would work in games but not in films.

As an art, writing is as complex as animation or programming. A great writer should be able to:

- Create complex and rich characters
- Simultaneously reveal a character's inner life while also advancing the plot
- Work symbols into a story to add emotional depth
- Hint at ambivalence and complexity in relationships between characters
- Create a multifaceted and layered world for the game
- Lace a theme, even a multifaceted one, into the story
- Lead the player through emotionally difficult situations and decisions that ultimately generate insight
- Incorporate other subtle components into a game that result in an emotionally broad and deep experience.

Utilizing only a single line of dialogue, a skilled writer should be able to create a fresh character, with deep emotions running beneath the surface of his or her words.

Can a game designer, artist, programmer, or producer with no professional writing background learn all this and more? Given talent, passion, and study, the answer is "probably," although it might take a few years and some abortive efforts along the way. Is learning all this the best use of his or her time for the needs of your project? I can't answer that.

Writers will certainly have a big role in the future of game design, but it will take a new kind of writer. Taking into account what has been suggested here and the increasingly collaborative nature of game development, it will likely be difficult to draw the line between where the craft of writing leaves off and where that of game design begins.

Until that day, if you're a developer who intends to hire an outside writer, it's up to you to get the most out of whomever you bring on board.

A SCREENWRITER HIRING CHECKLIST

Games with stories and characters need great writers, whether they come from screenwriting or some other arena. If you or your company is going to bring an external writer to your game project, here are a few things to keep in mind:

1. Don't get star-struck. Choose writers by their writing ability (that is, actually read their writing and decide what you think). Don't drop your jaw just because they're represented by a big agency. And even if you are using a "name" writer, be aware that often films and TV episodes go through many rewrites by writers who aren't credited.
2. Read a writing sample. Is this a great writer? Can he or she make both major and minor characters unique, interesting, dimensional, and emotionally rich? Is the story imaginative and gripping? Can this writer captivate you from beginning to end? I should mention that in Hollywood, people often write in teams. If the writing sample is from a team but you're only dealing with one of the writers, forget them. You have no way of knowing who wrote the best material in the script, no matter what the agent or writer claims. Similarly, many Hollywood scripts are rewritten many times, by many different writers. If you're shown a script and you're told that the writer wrote "this draft," the sample is useless. The agent and writer will protest and say the writer did a "page one rewrite," but in truth, you'll never have the faintest idea who created or wrote some of your favorite plot twists, characters, scenes, and dialogue.
3. Educate the writer in the points covered in this article, so he or she know why and how writing for games is different from other media.
4. Make the screenwriter prove that he or she can create emotionally complex NPC characters, some of which are likable and some of which are not, with very few words of dialogue.
5. Communicate the group nature of the creative process in games. Make sure the writer is in complete agreement with this process.
6. Use your gut to assess the screenwriter's motives. Does he or she actually care about games at all?
7. Look at their body of work. It's a plus if the writer has worked on other games. But for me, that wouldn't be enough. What if the game or games they worked on brimmed with disappointingly weak writing? Always insist on reading actual writing samples.

ACKNOWLEDGEMENTS

Thanks to Warren Spector, Gordon Walton, Richard Ham, Jason Della Rocca, Jeff Barnhart, Chris Klug, Dean Orion, Tyrone Rodriguez, Kenneth Holm, and Anand Rajan for their invaluable feedback during the writing of this article.

Monolith's No One Lives Forever 2: A Spy in H.A.R.M.'s Way

When creating a sequel to a critically successful first-person action-adventure game, it's not enough to measure up to the original. In order to meet fans' higher expectations, you have to surpass it.

Unfortunately, the dizzying rate at which game technology evolves means you'll probably be rewriting major engine components, such as your renderer or physics, as well as adding and revising game systems and overhauling tools and exporters. Your characters and environments will probably be more detailed, with more animations, more special effects, and more layers of complexity. In other words, you'll spend more time on less content.

There's also the problem of keeping people motivated. No one wants to spend a year and a half rehashing a game they just spent a year and a half developing. So you have to evolve the design sufficiently to excite the team, present new challenges, eliminate or rework elements of the first game you didn't like, and explore new gameplay concepts.

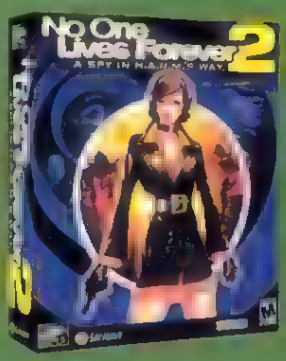
At the same time, you have to stay true enough to the essence of the previous game that you don't completely alienate your fan base. You'll find that many fans really want more of the same, only better, so you have to strike a balance between evolution and reiteration. Our mantra for the recently released No One Lives Forever 2: A Spy in H.A.R.M.'s Way (NOLF 2) was to create a game in the spirit of the original but not necessarily in its image. Judging by early reactions, we were fairly successful, although the game is certainly not without its controversies and shortcomings.

What Went Right

1. Identified core franchise elements. As proud as we were of the original No One Lives Forever (NOLF), released in late 2000 (for which I also wrote a Postmortem, available at www.gamasutra.com/features/20010608/hubbard_01.htm), we didn't want the sequel to be more of the same with different levels and new characters. The surest way to doom a franchise and alienate a team is to make decisions based on what was in the original game. Instead, we chose to navigate by what worked well in the original game.

We began by evaluating reviews and fan reactions to NOLF, comparing the general consensus to our own opinions, and charting out a course of action building on the first game's strengths without reiterating its weaknesses. It was imperative that we identify elements that fans liked and disliked, but it was also useful to identify which things no one noticed at all. Every feature in a game takes time to design, develop, refine, and test, so squandering precious days or weeks on anything that won't register with users is the last thing you want to do.

CRAIG HUBBARD | *Craig is the creative director at Monolith Productions. He was the lead game designer on No One Lives Forever 2: A Spy in H.A.R.M.'s Way, No One Lives Forever, and Shogo: Mobile Armor Division. He also worked as a level designer on the original Blood and contributed to Aliens vs. Predator 2. You can reach him at craig@lith.com*



GAME DATA

PUBLISHER: Sierra/Fox Interactive

NUMBER OF FULL-TIME DEVELOPERS
21 core team members with assistance from up to 4 other personnel at any given time

CONTRACTORS: Cinematic music scoring, motion capture actors, and voice actors

LENGTH OF DEVELOPMENT
19 months

RELEASE DATE:
Gold on September 19, 2002; available around October 4, 2002

TARGET PLATFORMS: PC

DEVELOPMENT HARDWARE: Pentium 1.0–1.7GHz machines with 256–512MB RAM and GeForce 1–4 video cards

DEVELOPMENT SOFTWARE: LithTech DEdit/ModelEdit, Microsoft Visual Studio (C++), Photoshop, Maya, 3DS Max

NOTABLE TECHNOLOGIES: LithTech Jupiter Development System

PROJECT SIZE: 2,000 files; 150,000 lines of code

The machine-gun toting mimes were some of the first character concepts created during the project

Based on our research, we determined the franchise's key elements were: a variety of interesting locales; memorable events, such as falling out of an airplane or being aboard a sinking cargo freighter; humorous conversations, documents, and characterizations; and an intriguing story told through cutscenes and in-game encounters. The franchise's flaws included a lack of visual polish; frustrating stealth elements; overly long, tedious cutscenes; and the inclusion of superfluous weapons and gadgets.

These summaries helped immensely in characterizing the essence of the NOLF franchise, but they also pointed out some fundamental problems we would face in creating the sequel. For example, given our schedule, it wasn't possible to produce the same variety of locales; because the content would be significantly more detailed, it would take longer to create. We also wanted a stronger, more deliberate visual presentation, which meant devoting more effort to each set. Furthermore, NOLF had made it clear that the more time spent building level geometry, the less time could be spent implementing gameplay.

Other issues were easier to address. NOLF had included roughly 30 weapons and gadgets, with a lot of overlap. For example, there were three pistols, two sniper rifles, several very similar missile-type weapons, and a couple of gadgets — such as the camera disabler and robotic poodle — that where rarely used. NOLF 2 has about the same number of items, but with a wider variety, including such devices as the Angry Kitty, banana, and bear trap, to compensate for the absence of the redundant firearms.

Our efforts to make stealth more intuitive and rewarding resulted in a redesign of the entire AI system. We decided that if we were going to allow players to sneak around and spy on their enemies, those enemies had better be doing interesting things. So Jeff Orkin, our AI engineer, and John Mulkey, our lead level designer, spearheaded the design of a goal system that would give non-player characters a sense of purpose, as well as a Smart Object system that would provide them with cues on how to interact with the environment.

The key to these systems is their unscripted nature. An NPC may start off working at a desk with a typewriter, get up and head over to the vending machine for a can of soda, step outside for a cigarette break, lean against a wall, walk over to a window to admire the view, and even run off to the restroom for a potty break. To ensure that players would be able to observe these behaviors, we effected a system that let us designate hiding places in levels from which you can watch enemies without being seen. Our implementation left something to be desired, but it still served the purpose of making stealth easier.

A more important refinement was to make it easier for players to escape from enemies. In NOLF, we used a time-based solution to determine whether an NPC would give up its pursuit, but it had lots of limitations and never worked very well. In NOLF 2, your ability to elude an enemy is based on passing through junctions. When an NPC reaches an intersection and doesn't see which way the player goes, it picks a course randomly. Level designers can weight a specific direction to make this decision-making seem more intuitive, so that an NPC is less

Bananas were a last-minute addition to the game but are useful for taking down enemies quietly

The goal-based AI system allows enemies to attack or flee from danger using smart objects within the world, such as these metal railings

FREE YOURSELF TO DO THE CODE THAT MATTERS

New for Intrinsic Alchemy 2.5

Features

- PC DLL Support
- New Plug-in Architecture for
  Insight Finalizer
  Samples Demos
- Embedded Viewer for Max & Maya
- Powerful Memory Allocation
  Logging & Debugging
- Max 5.0, Maya 4.5 Support

- Cross-Platform Self-shadow Shader
- Cross-Platform Glossmap Shader
  (specular map)
- Improved API Documentation

Benefits

- Easier Upgrades
- Easier to Integrate Custom Objects
  into Tool Chain

- Instant Game Content Preview
- Easy to Track Down Memory Leaks

- Even More Robust & Complete
  Content Coverage
- New Cross-Platform Shaders
- New Cross-Platform Shaders

With Intrinsic Alchemy, you're developing *games*, not tools.

INTRINSIC ALCHEMY

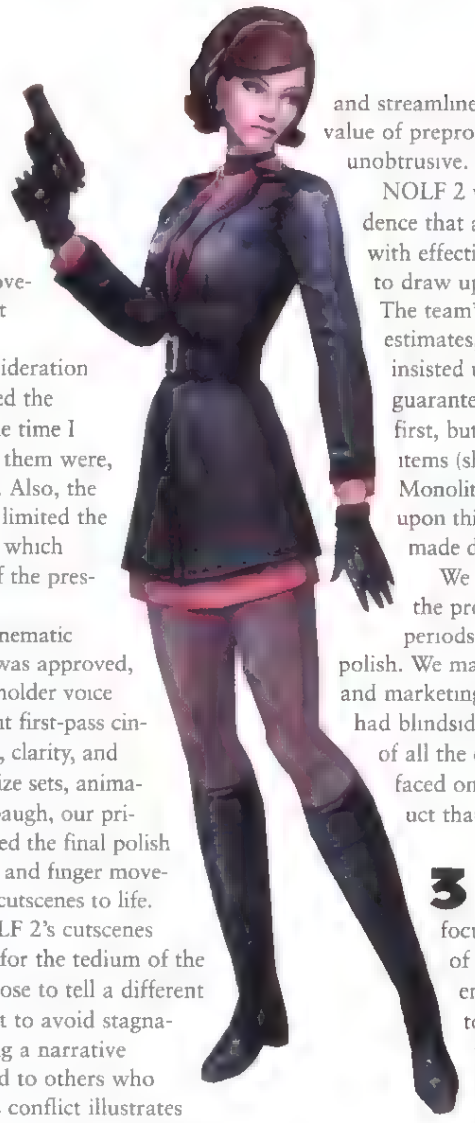
sales@intrinsic.com

Game Developer FRONTLINE FINALIST '02

INTRINSIC GRAPHICS

WWW.INTRINSIC.COM
650.966 6800

©2002, Intrinsic Graphics, Inc. All Rights Reserved. Intrinsic Alchemy and the Intrinsic Graphics logos are trademarks and/or registered trademarks of Intrinsic Graphics, Inc.



likely to investigate an alley than to continue down a major street. Designers can also specify what actions an NPC takes when it chooses a particular path. For example, if the NPC chooses to explore the alley, it may switch to slower, more tactical movement, whereas if it chooses the street, it keeps running.

Cinematics were another major consideration for the sequel. On NOLF, I implemented the cutscenes very late in the project. By the time I realized how long and tedious some of them were, it was far too late to do much about it. Also, the sheer number of cutscenes in the game limited the amount of time I could spend on each, which adversely affected the overall quality of the presentation.

By contrast, I finished the NOLF 2 cinematic script very early in the project. Once it was approved, we scheduled motion capture and placeholder voice sessions, which allowed me to implement first-pass cinematics that we could review for pacing, clarity, and continuity. Gradually, we began to finalize sets, animations, voice work, and pacing. Scott Albaugh, our primary character artist and animator, added the final polish with detailed facial animations and hair and finger movement. These touches really brought the cutscenes to life.

In retrospect, the conciseness of NOLF 2's cutscenes may have overcompensated somewhat for the tedium of the original's. Compounding the issue, I chose to tell a different type of story on this project in an effort to avoid stagnation, which alienated some fans wanting a narrative more in tune with NOLF's but appealed to others who wanted something new. Ultimately, this conflict illustrates one of the perils of creating a sequel. You have to evolve the franchise enough to keep it fresh, but not so much that you transform it entirely. It's a difficult balance to strike.

2. Preproduction phase and scheduling. After the debilitating chaos of NOLF's early months, we were determined to schedule a preproduction phase for NOLF 2 that would allow us to plan, prototype, and refine its core design before we started building publishable content or technology. The idea was that by the end of the preproduction period, we'd have a design that we could execute during production, refine during alpha, and test during beta.

Overall, preproduction was tremendously beneficial, with a shipping product remarkably faithful to the blueprint. Despite the inevitable setbacks and minor changes during production, our plan was solid enough to survive a complete rewrite of the renderer, new player physics, and various other technical and conceptual calamities. We remained flexible enough to revise and streamline as necessary, but it's a testament to the value of preproduction that such changes were largely unobtrusive.

NOLF 2 was completed on time and on budget, evidence that an AAA title can be developed in 18 months with effective scheduling. A detailed plan allowing us to draw up thorough, itemized task lists was the key. The team's experience enabled us to furnish realistic estimates, adding buffer time for insurance. We also insisted upon a prioritization system that not only guaranteed that critical features would be completed first, but also allowed us to jettison low-priority items (should time become an issue). Because both Monolith and our publisher, Fox Interactive, agreed upon this plan during the preproduction phase, cuts made during production were much less painful.

We were also smarter about how we scheduled the project. We allowed longer alpha and beta periods to ensure more time for play-testing and polish. We made sure to leave more room for E3, demos, and marketing and PR materials, as these interruptions had blindsided us on NOLF. The result was that in spite of all the obstacles and unforeseen challenges we faced on NOLF 2, we hit our ship date with a product that we're very proud of.

3. Upgraded tools. We knew early in preproduction that NOLF 2 would be more focused than its predecessor. Given the amount of extra detail that had to go into characters, environments, and objects, we couldn't hope to produce the same amount of content without forsaking quality, unless we streamlined the content creation and management pathways to let us work faster and more efficiently.

The single most important tool we added was the referential prefab system, allowing us to populate environments with objects that refer back to one original source file. In other words, edits made to one file propagate throughout the entire game. For example, if the sound department wants to add a sound to a door opening and closing, they only have to modify a single prefab instead of tracking down every single instance of that door in the game.

The primary advantage of this system is that it puts the power in the hands of the people who need it, without any programmer intervention. A level designer can create a block of geometry that represents a desk, with which he or she can plan the layout of a given room. The art team can then build a nicer-looking desk of roughly the same dimensions to replace the block. Level designers can hook up the drawers to open and close and add work nodes so that AIs can sit and type or fill out forms.

Other important improvements included robust exporters for

NOLF 2 required more complex special effects than its predecessor To accomplish this, a new FX editor was written at the project s beginning

Complex facial expressions and eye movemens were added by hand Lip synch files were generated by the Voice Works plug-in for Maya

3DS Max and Maya, which let us build and texture geometry in professional 3D packages and import it into our proprietary editor, where we added gameplay. Rather than enumerate every improvement, I'll just say that the decision to focus on upgrading our tools not only saved us immense frustration but also led to significantly higher quality content than we would have been able to produce otherwise.

4. Good team management. On NOLF, certain lead positions remained unfilled for months after the contract was signed. As a result, some of the most crucial people on the project arrived when it was already well underway. They inherited decisions that had been made without adequate expertise or experience, leading to redesigns, cuts, and inconsistent quality.

At the beginning of NOLF 2, we had great leads and good project management across the board. These key personnel were able to produce the detailed, organized documentation we needed in order to communicate with each other and our publishing partners. They were also able to provide accurate time estimates that led to a realistic schedule. The simple fact that we achieved most of what we set out to do without missing our ship date demonstrates the value of competent leadership.

The Angry Kitty explosive is a powerful albeit humorous addition to Cate s arsenal

5. Single-player and cooperative multiplayer synergy. The cooperative multiplayer component of the game proved challenging, but as we had hoped, it ultimately was far more complementary to the single-player game than competitive modes would have been. Much of what constituted the cooperative experience came directly from the single-player game, but some of the features we developed chiefly for cooperative play worked their way back to single-player and resulted in a better product. For example, we added the radar feature for co op play but quickly realized its value for solo play, which in turn led to tracking darts which allow players to mark enemy positions. In contrast, the competitive modes we added to the original NOLF really didn't benefit the single-player game at all. Furthermore, we found that traditional multiplayer divided our efforts, which resulted in a whole slew of new bugs, distracting the QA department.

In retrospect, we would have been smarter to add competitive multiplayer as part of the NOLF support package, which is what we're doing with NOLF 2. This approach allows us to focus our development and quality assurance efforts instead of diluting them across very different types of experiences.

What Went Wrong

1. Personnel issues. Building a solid, stable team was the most challenging hurdle we faced; in addition to replacing a couple of people who went to other projects, we needed to expand the team in a couple of key areas. The process of advertising available positions, screening poten-

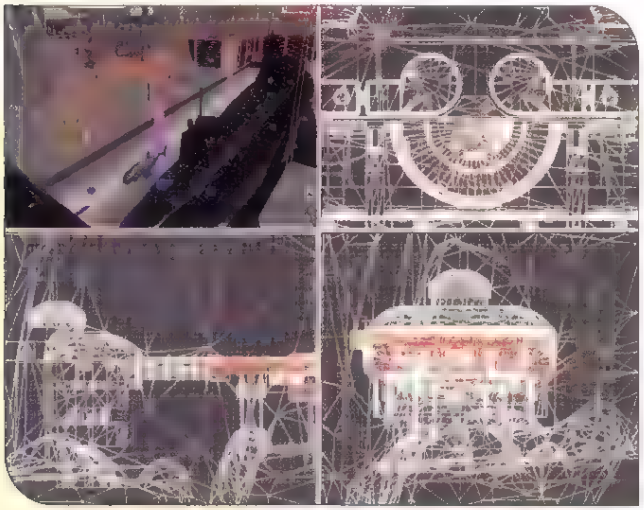
tial candidates, arranging interviews, and getting new employees on-site for training consumed enormous amounts of time. To complicate matters, our initial recruiting phase included several bad hires. In some cases, new employees proved to be less skilled than we had hoped; in others, poor attitudes or work habits led to dismissals.

Another major snag was that several team members, including me, had to help out on ALIENS VS. PREDATOR 2 for several months. The time we invested in that project was well spent, but NOLF 2 paid the price; the work we should have been doing on NOLF 2 had to be delayed until we returned. Given how many interdependencies there are on a project of this complexity, such setbacks can be frustrating and costly.

2. Preproduction phase too short. While preproduction was incredibly valuable for NOLF 2, it wasn't long enough. Ideally, we would not only have finalized the design but also fleshed out a solid playable prototype. We couldn't manage both in the three months available.

While our plan had been fairly thoroughly laid out, it hadn't been tested. This was our main problem. Therefore, any hitch we encountered affected the schedule. Needless to say, we ran into plenty of hitches and had to make lots of adjustments. By the time we shipped, we cut an entire mission from the game, numerous levels had been simplified, and our hopes of offering multiple solutions to every obstacle had been dashed (as I'll explain shortly).

3. Not enough iteration. Our goal was to have a solid plan by the end of preproduction that we would then execute during production and refine during alpha. In retrospect, we should have completed a rough draft of the game sooner in order to identify weak links and pacing issues while there was still plenty of time to address them. As it was, we ended up making some drastic adjustments very late in the project. For instance, we removed the entire exfiltration from Japan, which proved not to be as exciting as we hoped. In another case, we added an action-oriented level to break up a long section of slower-paced gameplay. These changes inarguably improved the game, but they should have been made earlier in the process.



In order to more quickly create high-detail content, DEdit, LithTech Jupiter's proprietary level-building tool, required significant upgrades

Also, while the game was decidedly more systemic than its predecessor, we still spent a lot of time tracking down individual items that could have been systematized. These cases usually involved components that couldn't easily be converted to prefabs. The windows in the main records building of the Siberian outpost were especially problematic, because they were all different sizes and shapes. If we wanted to change the amount of damage they could sustain before shattering or the radius of the resulting disturbance, we had to modify each one manually.

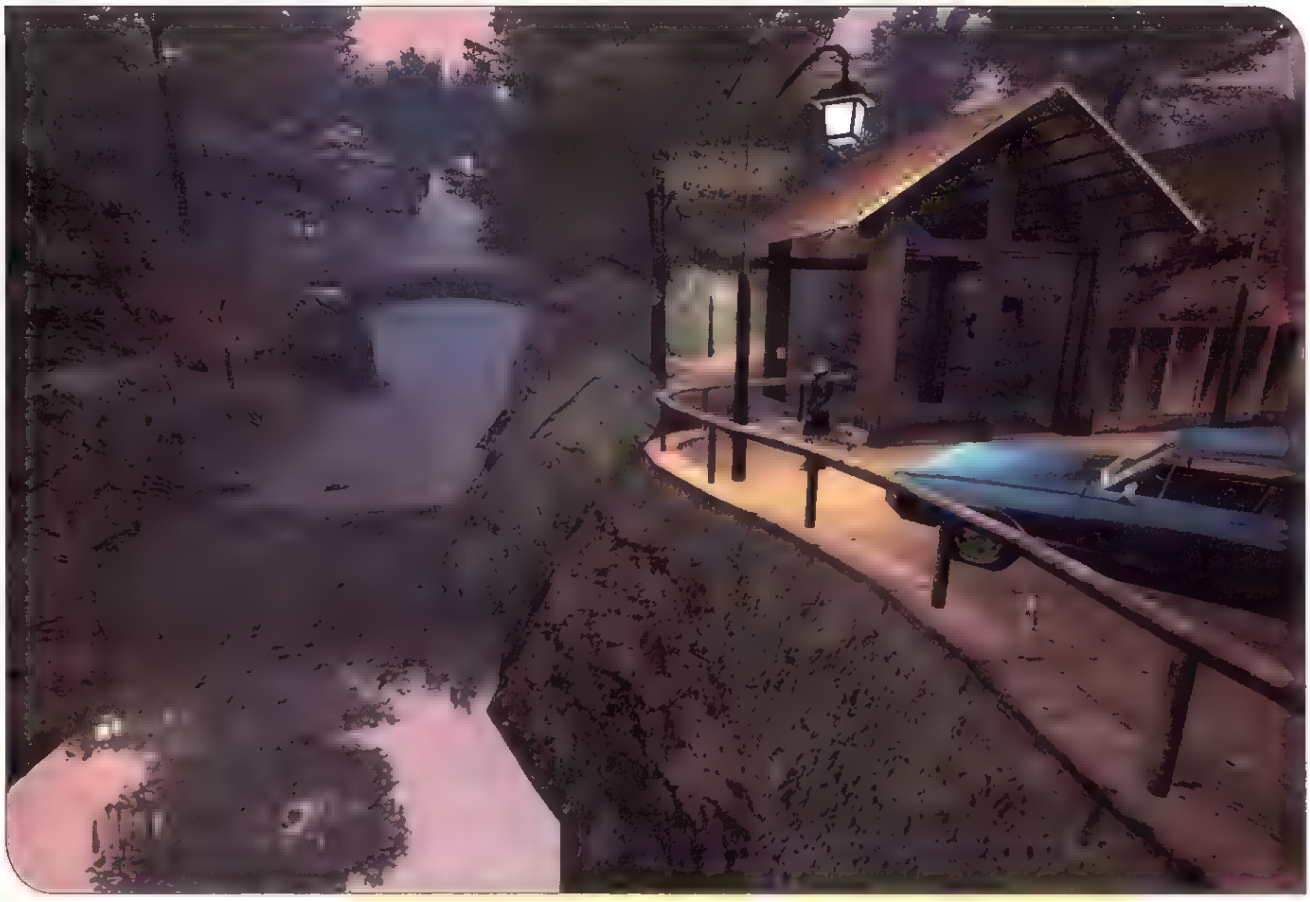
Finally, while play-testing helped balance and tune the game, it should have happened sooner. Thanks to observing playtesters, we made some crucial refinements to the stealth system and the opening missions, but we didn't have sufficient time to play-test the entire game. Play-testing also revealed some design flaws that couldn't be addressed without jeopardizing our ship date. While none of these issues was especially grave, they underscored the need to start play-testing as early as possible.

4. Crucial technology and tools lengthy to develop. During our analysis of NOLF, it became clear that the sequel would require significant graphical upgrades to compete in the rapidly changing PC game market. Upgrading our graphics required a rewrite of major engine components, such as the rendering pipeline. Because many of these features were not completed until after production had already begun, the team was occasionally forced to rework content as new technology came online.

The most dramatic change was the new renderer's occlusion system, which proved difficult to use and required a learning period for artists and level designers. During this time, the content team was in full production, meaning that environments were being created and detailed without a thorough understanding of the system's intricacies. Consequently, several layouts had to be significantly modified after a great deal of time and energy had already been invested in them.

5. Time constraints. While our scheduling was vastly better on NOLF 2 than on NOLF, we still crammed things too tightly for our own good. Unexpected budget cuts early in production ate up most of our buffer time. The aforementioned team issues took their toll as well. Most of us worked at least 100 hours per week during the last several months of the project, and some people were crunching even before that. In NOLF 2's closing credits, special thanks are given to Metrolabs, the developers of XTZ caffeine and herbal energy drinks, whose products sustained several team members in the absence of adequate sleep.

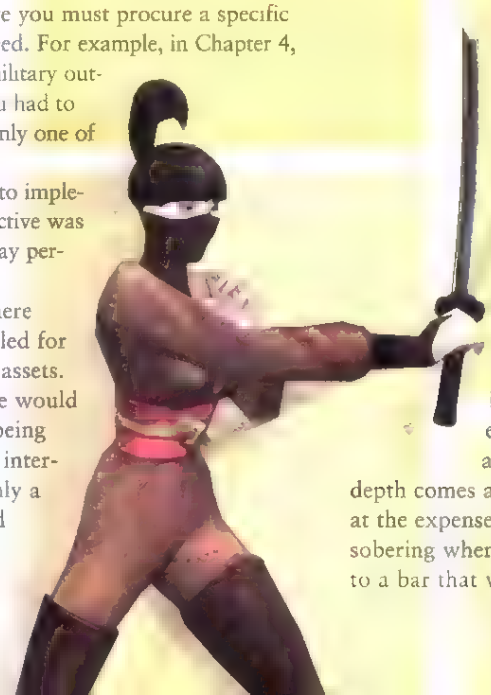
Time constraints led to some disappointing compromises. My most painful concession was giving up on our ambition to allow multiple solutions to every problem, which resulted in



Engine support for [illegible] snial stream or the nearby metal car with greater visual punch

various points in the game where you must procure a specific inventory item in order to proceed. For example, in Chapter 4, which takes place at a Siberian military outpost, there was a point where you had to get through an electrified gate. Only one of several planned options survived. Unfortunately, the easiest option to implement from a development perspective was the least desirable from a gameplay perspective.

Another problem was that there still wasn't enough time scheduled for public relations and marketing assets. We didn't anticipate that people would want so much from us. While being overwhelmed with requests for interviews and screenshots is certainly a great problem to have, it would have been better for everyone involved if we'd been more prepared.



The Price of Success

I concluded the original NOLF Postmortem by remarking that Monolith had matured from a disorganized but enthusiastic young company to a focused, professional business. NOLF 2 is the proof. Although the game is by no means perfect, it's a testament to the value of planning, organization, prioritization, and experience.

Games are more expensive to develop than ever before, but budgets rarely grow accordingly. Developers have to be smarter and more realistic in order to produce quality titles. It's important to realize that compromise is the essence of game development. Everything comes at a price: Polish comes at the expense of scope; depth comes at the expense of refinement; complexity comes at the expense of stability. These considerations are especially sobering when developing a sequel, as you must measure up to a bar that was set under very different circumstances.

Midway Home Entertainment Inc.
10110 Mesa Rim Rd. San Diego, CA 92121
email resumes to: resumes@midway.com fax: 858-658-9503
www.midway.com We are an equal opportunity employer.

©2002 Midway Home Entertainment Inc. The Midway logo is a trademark of Midway Amusement Games, LLC. Used by permission.

WE'RE BUILDING THE TEAM THAT WILL BUILD OUR COMPANY

Sammy Studios is looking for talented and creative people to break new ground. Help lay the foundation for a new, advanced video game development and publishing house under construction by Sammy Corporation, one of Japan's most successful entertainment companies.

Positions are available in our office and studio complex in Los Angeles, California, and our state-of-the-art 47,000 sq. ft. facility located in the coastal city of Carlsbad in sunny San Diego County, California:

Producer
Lead Artist
Lead Programmer
Lead Game Designer
Senior Programmer
Senior Game Designer
Senior Artist
Artist
Associate Artist

Senior Concept Artist
Concept Artist
Level Layout Designer
Programmer
Associate Programmer
Assistant Producer
Associate Producer
Sound Engineer
Tester

Join veteran artists, designers, producers and programmers from leading video game and entertainment companies. Create original, franchise-worthy titles for the current generation of video game consoles. Help lay the groundwork for the company and grow as Sammy grows into a global video game developing and publishing force!

Established in 2002 by leading industry professionals, Sammy Studios is a global developer, publisher and distributor of video games and entertainment content for a variety of platforms, including Sony PlayStation® 2, Microsoft Xbox™, Nintendo GameCube™ and Game Boy® Advance, PC and Macintosh. The company operates offices in the US and Europe, with its headquarters and a state-of-the-art development studio located in Carlsbad, California, and a second office and studio complex located in Los Angeles, California. Sammy Studios is a wholly owned subsidiary of Sammy Corporation of Tokyo, Japan, an entertainment and amusement conglomerate with over a billion dollars in annual revenue and one of the top-rated companies traded on the First Section of the Tokyo Stock Exchange. For more information on the company, please visit www.sammystudios.com <http://www.sammystudios.com/>.

Up-to-date listings and additional information are available at www.sammystudios.com/careers <http://www.sammystudios.com/careers>)

Sammy Studios

Savannah College
of Art and Design

It's Alive!, Alias/Wavefront Maya™ Adobe Photoshop® Deep Paint 3D™ and Adobe After Effects®
Terry Ziegelman, St. Louis, Mo. second year M.F.A.,
Paul George, Valencia, Venezuela. second year M.F.A.,
Stephen Johnson, Charleston, S.C., B.F.A.,
Jamie Kirschenbaum, Reston VA., B.F.A

BACHELOR OF FINE ARTS
MASTER OF ARCHITECTURE
MASTER OF ARTS
MASTER OF FINE ARTS

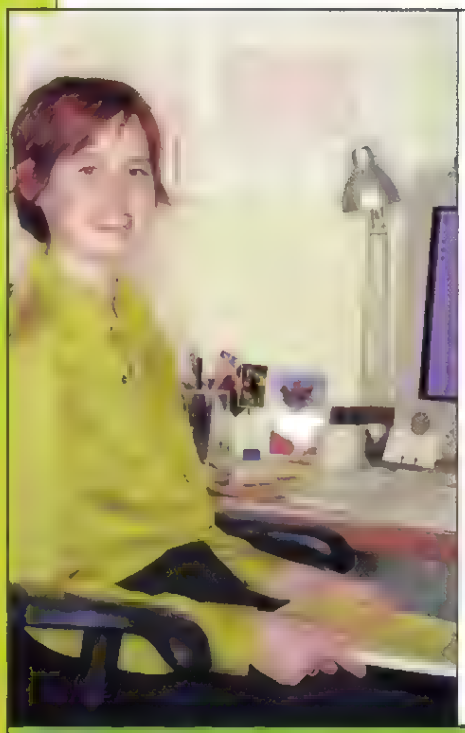
AN INTERNATIONAL UNIVERSITY FOR THE ARTS
Savannah, Georgia
800.869.7223 | www.scad.edu

ELECTRONIC ARTS™

Electronic Arts, a worldwide leader in interactive entertainment software, currently has openings for the following positions at our US headquarters in Redwood City, CA and at locations throughout the US, including studios in Los Angeles, CA; Walnut Creek, CA; Austin, TX; Maitland, FL and Las Vegas, NV

Computer Graphic Artist
Graphic Designer
Software Engineer
Web/Online Developer
Database Administrator
Programmer/Analyst
Online Producer

Note: For Computer Graphic Artist candidates, please include a demo VHS reel/disc with your resume. For complete details on these and other opportunities, submit your resume on the web at http://jobs.ea.com.; fax resumes to (650) 628-1387 or mail to: Electronic Arts, 209 Redwood Shores Parkway, Redwood City, CA 94065. Please reference job code: **GD-10/02**. EOE



Gamasutra.com

It's easy:

1 Just fill out a short form about your services and area of expertise, and you'll go up on the site the same business day.

2 **Basic listings are FREE**, and enhanced listings that let you upload logos, work samples and more are just a couple of dollars per day.

GamaNetwork

Game contractors: get listed.
www.gamasutra.com/getlisted/

Computer Graphics

- 2D & 3D Animation
- 3D Modeling
- Character Design
- Computer Arts
- Computer Illustration
- Digital Imaging
- Digital Media
- Digital Photography
- Gaming
- Visual Effects
- Web Design
- And More...

AA | BFA | MFA Degrees | Portfolio Development | Online Classes

Apply Now for Spring, Summer & Fall Semesters

High School Scholarships & Teacher Grants Available

AcademyOfArtCollege

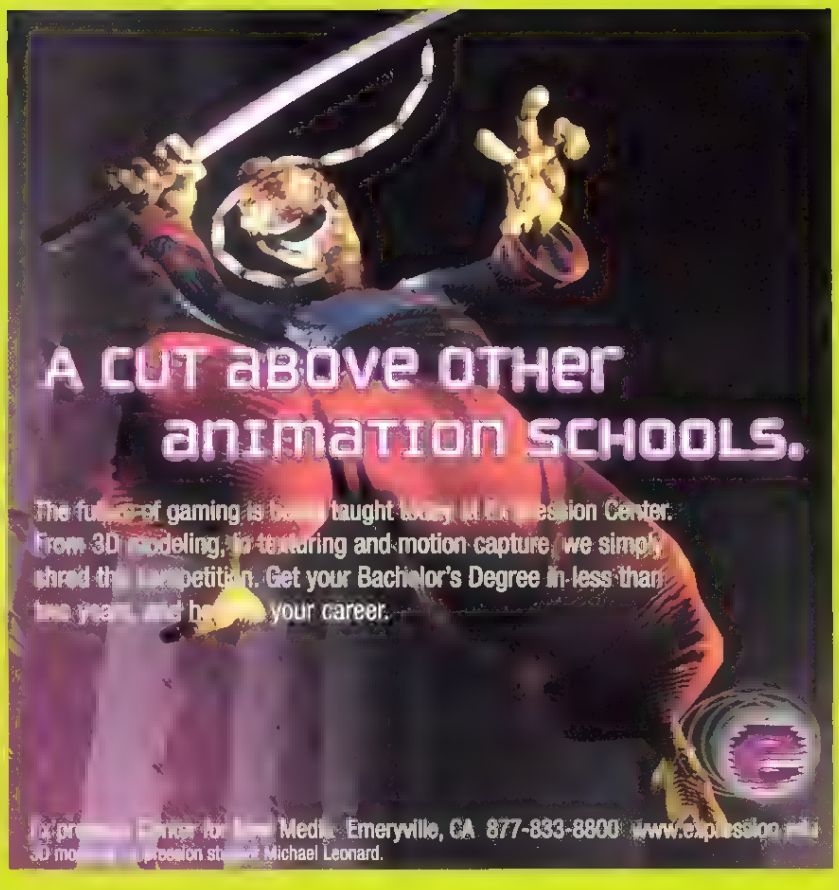
1.800.544.ARTS | www.academyart.edu

79 New Montgomery St. | San Francisco, CA 94105 | Nationally Accredited by ACICS, NASAD & FIDER | Established in 1929

THINK YOU COULD ROCK THE TECH WORLD?

Prove it.

UAT



Michael Leonard.

If you're going to pay for an orange...

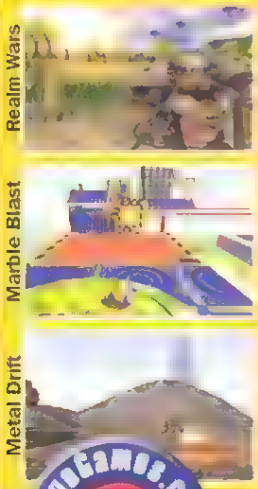
you should get an orange.

If you squeeze the new DVD from VFS you will never again be confused about who's got the juice.

To get more information or your own DVD of VFS student work call 1-800-661-4101 or email dvd@vfs.com

Vancouver Film School
Creative. Disciplined. Focused.

Invest in Your Own Game!



- Award-Winning Network Code
- Engine and Tool Source
- Supports PC, Mac and Linux Platforms
- Seamless Indoor & Outdoor Render Engine
- Powerful Scripting Language

$100 per programmer seat for Indie developers. Commercial licenses available with NO royalties. The **Torque Game Engine** is supported and expanded by one of the world's largest communities of independent developers.

TORQUE GAME ENGINE

GarageGames

www.garagegames.com

345 Mill Streeet, Suite #200 • Eugene, OR 97401 • 541.345.3040

FULL SAIL

Real World Education

School of:

Game Design
Computer Animation
Digital Media
Film
Audio
Show Production

800.226.7625
www.fullsail.com

3300 University Boulevard
Winter Park, FL 32792

- Financial aid available to those who qualify.
- Job placement assistance.
- Accredited by ACCSCT.

©2001 Full Sail, Inc. All rights reserved.
The terms "Full Sail," "Full Sail Real World Education," and the Full Sail logo are either registered service marks or service marks of Full Sail, Inc.

nPOWER

NEW 3DSMAX PLUG-INS!

Better Tools to Save You Time & Money!

Empowering Creativity with Advanced Technologies

POWER BOOLEANS

Power Booleans is a MAX [illegible] extreme [illegible] MAX modifier.

Fewer Polygons & Cleaner Meshes.
Fully Animatable Boolean Operations!
Improved Usability with Multi-Object Selection.
Integrated Decimation Polygon Reduction!

POWER SOLIDS

nPower Solids introduces a traditional CAD Boolean [illegible] solid into 3DSMAX.

View Dependent Tessellation for Rendering.
Import IGES, Rhino,STEP & SAT files.
Powerful Construction Primatives.
Filleting, Shelling & Booleans!
Animation of all Solids Operations.

www.npowersoftware.com
(858)-538-3800

i2i
ANIMATION
ART, STAGE & TECHNOLOGY... EYE TO EYE
CONCEPT
RIGGING
MODELING
ANIMATION
i2i ANIMATION, INC.
(503) 368-3067
WWW.I2IANIMATION.COM

·MUSIC
·SPEECH ·SOUND
·AUDIO INTEGRATION
OMNI Interactive Audio
"This team is made up of people who I know have the knowledge, experience and attitude to produce some of the best audio in the business. They have created amazing audio content for great games in the past and I'm confident will continue to maintain a "state of the art" capability in the future."
Mike Ryder President, Sierra Entertainment
"Alistair Hirst is not only a long time industry vet, but is also one of the great innovators that has worked to push the envelope of interactive music and sound on virtually every piece of gaming hardware over the past twelve years."
Chris Taylor - President, Gas Powered Games
We needed some extra help on sound effects for Sim City 4 to make Alpha, and Omni Interactive Audio delivered high quality work, on spec, with a quick turn around. I'd recommend them to any in-house team that are in a crunch.
Jerry Martin – Audio Director, Maxis
www.OmniInteractiveAudio.com

...we make models
...we make textures
...we make animation
NEW PENCIL
WE MAKE ART
80 Liberty Ship Way, Suite 6 Sausalito, CA 94965
phone 415.339.1800 fax 415.339.1803
web www.newpencil.com

continued from page 64

In both cases, many social critics saw the make-believe as promoting or inspiring the phenomena it portrayed, while the makers of the make-believe, perhaps too caught up in the process of fantasy themselves, seemed incapable of analyzing or articulating their full relationship with their customers. Time has shown that filmgoers in fact worked out sexual understandings far more complex than the Hollywood model, and the generations that began to reject warfare did so despite the armies that filled their toy chests. We can understand how that was when we understand how and why we engage in fantasy play.

What's new about electronic games is that no technology has ever before allowed teenagers and adults to immerse themselves as deeply as children do in such narrative, reality-based play. This isn't violence in the abstract form of chess or with the controls of football or viewed through the distance of an action movie, nor is it the all in the-mind's-eye violence of little kids. This is new territory for fantasy play.

Whether we like it or not, games have been given a difficult but important cultural mission. They make us ask why so many good kids want to play at violence, and why so many grown-ups want to keep playing in fantasy land. If we open ourselves to this discussion, we'll learn a great deal about what play, and what violence, means to us all. The answers will help us create more satisfying games — and may also help create a happier and less violent world.

GERARD JONES | *Gerard is the author of* Killing Monsters: Why Children Need Fantasy, Superheroes, and Make-Believe Violence *(Basic Books).*

ADVERTISER INDEX

COMPANY NAME	PAGE	COMPANY NAME	PAGE	COMPANY NAME	PAGE
3Dlabs	17	Garage Games	61	Perforce	7
Academy of Art College	59	Havok	5	Premier Press	31
Analog Devices	29	i2i Animation	62	Programmer's Paradise	9
Charles River Media	24	Integrityware	61	RAD Game Tools	C4
Compaq	13	Intrinsic Graphics	51	Sammy Studios	57
Conitec	63	Metrowerks	19	Savannah College of Art and Design	58
Criterion Software	2–3	Microsoft	15	University of Advanced Computer Technology	60
Discreet	C2	Midway Home Entertainment	56	Vancouver Film School	60
Electronic Arts	58	New Pencil	62	WildTangent	32–33
Expression Center for New Media	60	Numerical Design Ltd	10	Zona, Inc.	21
Full Sail Real World Education	61	Omni Interactive Audio	62		

Info, sales, free demo, updates, support: Visit www.3dgamestudio.com
"An excellent toolkit to quickly prototype and develop 3D Graphics applications."
Dr. Dobb's Journal - August 2000.
With TERRAIN SYSTEM support
3D GAMESTUDIO
VERSION 5
Powerful Features ! flexible, accessible scripting language
Contains World & Model Editors, A5 Engine / 6 degrees of freedom Mip & Shadow Mapping / Particles, Flare Effects, Fog / Static and Dynamic light sources / 3D Sound / Runs in D-3D & Software Mode Blazing Fast even w/o 3D Hardware / 2D Engine for Menus, Sprites and Text / Scripting language for AI & User Interface / Import Multiple Format: 3DS, ASC, MDL, MAP, BMP, PCX, WAD, MID.
...TEC DATASYSTEMS, Inc. 1951 4th Ave. Suite 301, San Diego, CA 92101. Tel: 619-702-4420
Standard Edition: $49.00 / EXTRA Edition: $99.00 / Commerc. Edition: $199.00 / Prof. Edition: $1,250.00 (Incl.: Client/Server IPX + TCP/IP, AVI + CD Audio) / Source Dev. Kit Available.

Playing It Out

Johnny likes to kill. First it was bad guys in his bedroom, dispatched with a finger that magically became a gun, later it was armies of evil droids in the backyard so fearsome that they required an alliance of friends armed with Nerf guns.

Now that he's too old to run around making shooting sounds, he turns to the enemies that invade his TV screen: zombies and aliens and sometimes, when a wicked mood is on him, the police who try to stop him from stealing cars.

We try to understand interactive games by comparing them either to existing narrative forms like movies or to more abstract, cognitive pursuits: chess, puzzles, and training systems. So we ask questions like, "What's this game teaching?" Or, "What's it glorifying?" But the answers are never satisfying and often needlessly alarming — because the questions are wrong. A better question is, "What is this game helping us play through?" The way to understand the most visceral, immersive, and transgressive games is as extensions of the fantasy play of childhood.

The anthropologists and psychologists who study play tell us that it has many functions, including mastery of cognitive skills and rehearsal for who we will be in later life. But one of its most important functions is harder to fit into our idealized image of childhood and so is less often discussed: allowing an escape into an alternate world in which kids can safely experience powers and freedoms that the restrictions of real life do not allow.

Some of those restrictions are physical. Children are small and vulnerable, so in their play they become big and tough. Others are emotional. We require them constantly to be "good kids," held to standards of self-control and prosocial behavior onerous even to the best-adjusted of them. In many ways we constrain our kids less than generations past. But there is one arena in which we in the contemporary West restrict children's impulses as no society ever has before: aggression.

No society has ever pounced so anxiously on fighting and yelling and every spontaneous expression of anger, has ever demanded so rigorously of children that they suck in their hottest feelings, manage their conflicts, and not provoke anyone. No society has ever brought such disapproval, fear, and white-knuckled control even to play-fighting and make-believe mayhem. We make it inevitable that children will need to explore fantasy worlds full of the aggression that inspires such anxiety, when real life allows so little chance to understand.

Those restrictions don't vanish with childhood, of course. All through adolescence and adulthood we're still required to be "good kids," which now more than ever requires the denial of aggressive thoughts. It's no accident that kids', teens', and adults' appetites for violent fantasy in movies, music, and TV are so high. And it's inevitable that the newer plaything, the electronic game, would have to be significantly concerned with violence from the start.

This isn't the first time a new play or entertainment technology has coincided with a mass upwelling of psychological concerns. Movies rose at a moment of conflict and anxiety about sexual mores, and so fantasy sex became one of early cinema's central concerns. In the 19th century, the technology to mass-produce cheap metal toys coincided with the rise of the militarized nation-state, with its centralized regimentation of the populace, and so the dominant product of the new toy industry because its stamped-out legions of identical soldiers.

Illustration by Jeff Spokes

continued on page 63

THE 3RD ANNUAL GAME DEVELOPERS CHOICE AWARDS

Nominations open January 2nd, 2003

"My passion is games, and having my life's work recognized by developers worldwide is the most meaningful honor I could receive."

– Yuji Naka, Sega
Lifetime Achievement Award recipient

game developers choice awards

Started in 2000 with the idea that there is no greater honor than to be recognized by one's peers, the Game Developers Choice Awards are game development's most prized honors.

All professional game developers are eligible to nominate the developers who transcended the state of the art in 2002 at **www.igda.org/awards**.

Sponsored by:

RenderWare

Marty O'Donnell
Excellence in Audio recipient for HALO

Tetsuya Mizuguchi
Game Innovation Spotlight recipient for REZ

Fumito Ueda
Excellence in Level Design, Excellence in Visual Arts, Game Innovation Spotlight recipient for ICO

Bob Rafei
for DAXTER, Original Game Character of the Year

The Best in Game Development Technology

Pixomatic Rendering Technology

Now you know what Michael Abrash and Mike Sartain have been up to! Designed for today's games, **Pixomatic** is their new software renderer that provides MIP mapping, bi-linear filtering, alpha-blending, alpha-test, point sprites, multi-texture, 32-bit and palettized textures, **DOT-3 bump mapping**, fog, specular, **stencil and z-buffering**, indexed primatives, multiple streams, high-speed blitting, and more! Pixomatic uses runtime code generation to create **optimized MMX, 3DNow!, and SSE** code on-the-fly! You will be amazed at its speed (what else would you expect from everyone's favorite optimizer?)! **Target the mass market** - don't let ancient 3D cards, buggy 3D drivers, or lack of 3D hardware prevent people from playing your games!

Bink Video Technology

The very best video codec - make your videos shine! Unrivaled quality (**better than DVD** and MPEG II). Up to **three times faster** than other true-color codecs. Perceptibly lossless, **8 to 1 audio compression**. Decompresses to DirectDraw, DIBSections, YUV overlays, or any other memory. Available for Win32, MacOS, Xbox, and now **Nintendo GameCube**. Think Bink!

Granny 3D 2.2

Granny is a **sophisticated dynamic 3D animation system** with an optimized run-time engine that delivers incredible performance in a tiny memory footprint perfect for consoles, PCs and Macs. **MAX & Maya** plug-ins export materials, multiple vertex UVs and colors, mesh data, and complete animation and skinning information. Granny also now includes **amazing normal map generation** - give Granny a high and low-res model and she'll build the normal map! These features, plus advanced camera control, collision detection, animation and texture processing, MIP-map generation, and **Bink and S3TC** texture compression, all add years worth of features to your game in just a few hours of integration time!

Miles Sound System

The ultimate sound system for PC and Mac! Miles supports 2D and 3D audio, MIDI with DLS, streaming, CD audio, **DSP filtering**, **MP3 playback (patent rights included)**, **Internet voice chat** (2900, 2400, and 1200 bps codecs), Creative **EAX 1, 2 & 3**, Aureal **A3D 1 & 2**, RSX 3D, DirectX 7, Dolby Surround, QSound, **super-fast software EAX emulation**, and more!

© GAME TOOLS

425 893 4300
www.radgametools.com

POWERFUL TECHNOLOGY. EASY TO USE. NO ROYALTIES.

Made with love by

Our goal is to preserve classic video game magazines so that they are not lost permanently.

People interested in helping out in any capacity, please visit us at retromags.com.

No profit is made from these scans, nor do we offer anything available from the publishers themselves.

If you come across anyone selling releases from this site, please do not support them and do let us know.

Thank you!